COMPUTER SYSTEMS LABORATORY

STANFORD UNIVERSITY · STANFORD, CA 94305-2192

89-0323

AD-A221 804

THROUGHPUT PERFORMANCE OF A DIRECT SEQUENCE CDMA PACKET RADIO NETWORK

James S. Storey and Fouad A. Tobagi

SEL Technical Report No. 85-277

SURAN Temporary Note No. 20

DTIC

June 1985

This work was supported by the Defense Advanced Research Projects Agency under Contracts MDA 903-79-C-0201 and MDA 903-84-K-0249.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER 85-277	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) THROUGHPUT PERFORMANCE OF A DIRECT SEQUENCE CDMA PACKET RADIO NETWORK		5. TYPE OF REPORT & PERIOD COVERED Technical Report
		6. PERFORMING ORG. REPORT NUMBER 85-277
7. AUTHOR(s) James S. Storey and Fouad A. Tobagi		8. CONTRACT OR GRANT NUMBER(s) MDA 903-79-C-0201 MDA 903-84-K-0249
9. PERFORMING ORGANIZATION NAME AND ADDRESS Stanford Electronics Laboratory Stanford University Stanford, California 94305-2192		10. PROGRAM ELEMENT PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Defense Advance Research Projects Agency Information Processing Techniques Office 1400 Wilson Blvd., Arlington, VA 22209		12. REPORT DATE June 1985
		13. NUMBER OF PAGES 95
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Resident Representitive Office of Naval Research Durand 165 Stanford University, Stanford, CA 94305-2192		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release; Distribution Unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A Code Division Multiple Access (CDMA) spread spectrum packet radio network model is presented. The topology is a single-hop fully connected network with identical users. The network model allows the performance of the radio links to be specified in detail. A model of a BPSK Direct Sequence (DS) CDMA radio channel with convolutional Forward Error Correction (FEC) coding is used. A refinement of the network model takes into account the restriction that half-duplex radios cannot transmit and receive simultane-

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

S N 0102-LF-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

ously. Single hop throughput and probability of successful first transmission are derived. The effects on throughput of the received signal strength E_b/N_0, the number of chips per bit N and the number of users M are shown. A channel load sense access protocol is introduced in which radios are blocked from transmitting when the channel is heavily loaded. The increase in throughput due to this protocol is shown for zero and for non-zero propagation delay.

Throughput Performance of a Direct Sequence CDMA Packet Radio Network*

SEL Technical Report No. 85-277

SURAN Temporary Note No. 20

June 1985

James S. Storey and Fouad A. Tobagi
Computer Systems Laboratory
Department of Electrical Engineering
Stanford University, Stanford, CA 94305-2192

Abstract

A Code Division Multiple Access (CDMA) spread spectrum packet radio network model is presented. The topology is a single-hop fully connected network with identical users. The network model allows the performance of the radio links to be specified in detail. A model of a BPSK Direct Sequence (DS) CDMA radio channel with convolutional Forward Error Correction (FEC) coding is used. A refinement of the network model takes into account the restriction that half-duplex radios cannot transmit and receive simultaneously. Single hop throughput and probability of successful first transmission are derived. The effects on throughput of the received signal strength E_b/N_0, the number of chips per bit N and the number of users M are shown. A channel load sense access protocol is introduced in which radios are blocked from transmitting when the channel is heavily loaded. The increase in throughput due to this protocol is shown for zero and for non-zero propagation delay.

*This work was supported in part by the Defense Advanced Research Projects Agency under Contracts MDA 903-79-C-0201 and MDA 903-84-K-0249, monitored by the Office of Naval Research.

case for half-duplex radios.

Earlier CDMA network models assume a network access protocol which is analogous to ALOHA; namely, if a packet arrives at an idle radio it is immediately transmitted. For ALOHA systems, network performance can usually be improved by introducing carrier sensing, where the users listen before transmitting and block transmission if another carrier is sensed. In this report, we investigate the improvement that is realized by using a channel load sense access protocol in a CDMA network. In this protocol, radios are allowed to begin transmitting a packet only if the total number of transmissions on the channel is less than a threshold K.

We analyze specific fully connected configurations of a more general multi-hop model developed by Boorstyn and Kershenbaum [BOOR80] and also by Brazio and Tobagi [BRAZ84]. For these configurations, we find the single-hop throughput and the probability of correct packet reception. The models are for asynchronous networks with variable length packets that are generated by Poisson processes. In Section 2, we present the system assumptions underlying the models analyzed. In Section 3, we present a simplified network model, which assumes a simplified channel model. We analyze the throughput and the probability of a packet being successful, and present some performance results. In Section 4, we present a more accurate channel model which can include forward error correction (FEC) coding. We refine the network model to accomodate this channel model. In Section 5, we modify the network model in order to investigate the performance of a network of half-duplex radios. Then, in Section 6, we add channel load sensing to the models developed earlier, and investigate the resulting improvement in performance. By using an approximate model of propagation delay, we find the degradation in throughput that this delay causes in networks using channel load sensing.

2. System Assumptions

The general model applies to various networks with the following common features.

1. Introduction

For many years, various forms of spread spectrum signalling have been used in secure communications systems. Though primarily implemented in military systems because of their anti-jam and anti-intercept properties, spread spectrum techniques offer other capabilities such as ranging and code division multiple access (CDMA). The performance of CDMA systems which use forward error correction (FEC) coding has been analyzed by various authors. Raychaudhuri [RAYC81] analyzed throughput, delay, number of retransmissions and stability of a slotted single hop network. Musser and Daigle [MUSS82] derived the throughput for a single hop unslotted network with fixed length packets. Pursley [PURS83] studied asynchronous frequency hopping systems with fixed length packets, to find the throughput and the probability of a packet being correctly received. Taiple [TAIP84] investigated the channel and network performance of a Direct Sequence CDMA (DS-CDMA) packet radio system.

Previous models and analyses of the performance of CDMA systems fall into one of two groups. The first group focuses on the radio links, deriving bit error rates (BER) for a given code and number of interfering transmissions. The other group focuses on network performance, deriving throughput and delay. Typically, in the analyses of network performance, a very simple channel model is used to approximate the radio link performance. Also, the models only account for the transmitters, assuming that a receiver is always available.

In this report, we present a network model which is flexible enough to incorporate a variety of channel models for the radio links. We than adopt a channel model of a Direct-Sequence Binary Phase Shift Keying (DS-BPSK) radio link presented by Taiple [TAIP84], and find the network throughput. A modification of the network model allows us to account for the receivers as well as the transmitters. We investigate the throughput of a network in which radios cannot transmit and receive simultaneously, as would be the

The network is fully connected with only single hop traffic. All users are identical. In Section 5, we use an M user finite population model, and consider each user to have a directed link to each of the $M-1$ other users. We assume equal traffic requirements on all $M(M-1)$ of these links.

The network has the following characteristics due to the use of spread spectrum CDMA. Radios are assumed to use receiver directed codes. The destination receiver has a probability α_ℓ of synchronizing to the packet header (capturing the packet), where α_ℓ is a function of the number of radios transmitting at the start of the packet transmission. Once capture occurs, we assume that the receiver never loses synchronization until the packet is completed.

Two channel models are considered. The first is a simple model in which no transmission errors occur for L or fewer simultaneous transmissions, and transmission errors occur with probability 1 for greater than $L+1$ simultaneous transmissions. This is referred to as the step function channel model, as the probability of bit error vs. the number of interfering transmissions is a step function.

The second channel model is a more accurate model of a DS-BPSK CDMA system. Channel performance results are found for a system incorporating FEC coding as well as for an uncoded system. The type of coding modeled is convolutional coding with Viterbi decoding. In our analysis, we assume an upper bound L, such that more than L simultaneous transmissions will cause all packets to be unsuccessful with probability 1. Of course, there is a small probability that a packet will be successful even if the number of interfering transmissions is so great that the probability of bit error is nearly 1/2. Nevertheless, the time averaged contribution to throughput by such packets is very small. We choose the cutoff point L such that the error due to ignoring larger numbers of interfering transmissions is negligible.

In both channel models, we assume that a single error in the corrected information

bit stream will cause the packet to be unsuccessful. In order to identify packets with bit errors, parity or cyclic check bits would be appended, and those packets found to be in error would be discarded. We will ignore the overhead this parity check would require.

We ignore the effect of acknowledgments, assuming a perfect and instantaneous acknowledgement channel is available.

A packet which is scheduled for transmission is blocked (i.e., rescheduled) if the radio is currently transmitting or receiving. In the channel load sense models, the transmission is also blocked if the number of radios currently transmitting is greater than or equal to a threshold K. Packets that are not blocked are immediately transmitted at the scheduling point

For channel load sense models, it is assumed that each radio has perfect knowledge of the number of radios currently transmitting. At a scheduling point, the decision whether or not to transmit is based on this knowledge. In a real system, the users can only estimate the channel loading. For example, if headers are transmitted using a general code rather than a receiver directed code, the user can count the headers received in the recent past to estimate the number of other users transmitting. Alternatively, the received signal plus noise power can be integrated to give an estimate of the total energy in the pseudo-noise signals received, which will indicate the channel loading. Clearly, these implementations will perform worse than the idealized model.

3. Network Model

In the basic model, we assume that receivers are dedicated to transmitters, so that transmissions are never blocked due to the source being busy receiving, and transmissions are never unsuccessful due to the destination being busy. For finite transmitters, each user generates scheduling points from a Poisson process at the rate of offered traffic λ. This Poisson traffic includes retransmissions as well as newly generated packets. We

use the heavy traffic assumption that there is a packet available for transmission at every scheduling point. If the user is not already transmitting and the protocol does not inhibit transmission (in the channel load sense protocol), the transmission of the packet will begin at the scheduling point. For an infinite population, the scheduling points for all transmissions are generated from a Poisson process with aggregate rate λ. Again, a packet will begin to be transmitted at the scheduling point if the protocol does not inhibit transmission. The transmission time of the packets is exponentially distributed with mean $1/\mu$. We use g to denote the normalized rate of offered traffic, $g = \lambda/\mu$.

In this simplified model, we assume a channel behavior such that no errors occur for L or fewer radios transmitting, and errors occur with probability 1 for more than L simultaneous transmissions. A more realistic model of the channel behavior is presented in Section 4.

Let $\mathcal{X}(\tau)$ be the number of radios transmitting at time τ (We use τ to denote time throughout the report to avoid confusion when we use t as the number of active transmitters in Sections 5 and 6). We find that $\mathcal{X}(\tau)$ is a continuous time Markov chain. The state space $\mathcal{S}$ of $\mathcal{X}(\tau)$ is $\mathcal{S} = \{0, 1, 2, ...\}$ for an infinite population, and $\mathcal{S} = \{0, 1, 2, ..., M\}$ for finite transmitters. In both cases, $\mathcal{S}$ has a strictly positive stationary distribution $\{\pi_j, j \in \mathcal{S}\}$. We notice that $\mathcal{X}(\tau)$ for the infinite population model has the same state transition rates as the model of the $M/M/\infty$ queue, shown in Fig. 3.1, and for the finite transmitters case, the same as the $M/M/\infty//M$ queue, shown in Fig. 3.2 [KLEI75]. For an infinite population,

$$\pi_j = \frac{(\lambda/\mu)^j}{j!} e^{(-\lambda/\mu)} \qquad j = 0, 1, 2, ... \tag{3.1}$$

For finite transmitters,

$$\pi_j = \frac{(\lambda/\mu)^j}{(1 + \lambda/\mu)^M} \binom{M}{j} \qquad j = 0, 1, ..., M \tag{3.2}$$

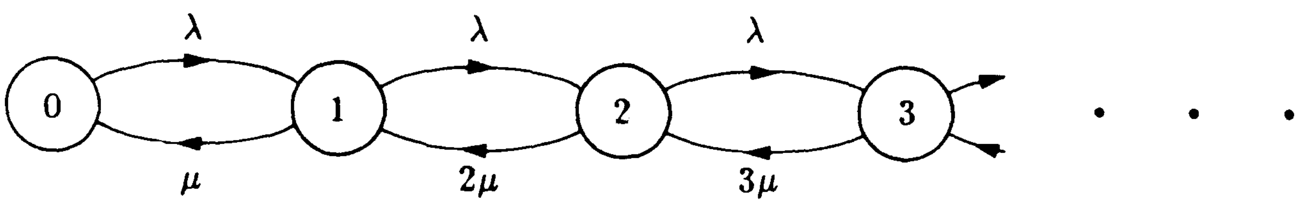

Figure 3.1. State-transition-rate diagram for the infinite population model.

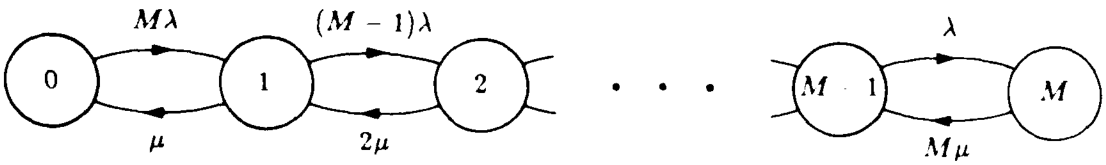

Figure 3.2. State-transition-rate diagram for the finite transmitters model.

3.A Probability of Success

We derive the probability of success for the infinite population model by tracing the evolution of the network model during the transmission of a tagged packet. If the transmission is completed before the state $L+1$ is reached, the packet will be successful. Conversely, if the state $L+1$ is visited first, the packet will be unsuccessful.

We define the following parameters:

User i is the source of the tagged packet

τ_0 is the time at which the transmission of the tagged packet begins

P_S is the probability that a transmitted packet is successful

$P_{S|j}$ is the conditional probability that a transmitted packet is successful given that the network was in state j at time τ_0^-

$\tilde{\pi}_j$ is the steady state probability of the network being in state j at time τ_0^- given that a packet transmission begins at time τ_0

$P_{i_{IDLE}}(j)$ is the probability that a tagged user i is not transmitting given that the network state is j

To find P_S, we first condition on j.

$$P_S = \sum_{j=0}^{\infty} \tilde{\pi}_j P_{S|j} = \sum_{j=0}^{L-1} \tilde{\pi}_j P_{S|j} \tag{3.3}$$

The tagged packet began transmission at time τ_0, which implies that user i was idle at time τ_0^-. Thus,

$$\tilde{\pi}_j = \Pr(\, j \text{ transmitters} \mid \text{user } i \text{ was idle})$$

For an infinite population, due to the Poisson generation process, $\tilde{\pi}_j = \pi_j$. For finite transmitters, we find $\tilde{\pi}_j$ from Bayes' Theorem as

$$\tilde{\pi}_j = \frac{P_{i_{IDLE}}(j)\pi_j}{\sum_{j=0}^{M-1} P_{i_{IDLE}}(j)\pi_j} \tag{3.4}$$

Because all users are identical,

$$P_{i_{IDLE}}(j) = \frac{M-j}{M} \qquad \text{for } i = 1, 2, .., M \tag{3.5}$$

Thus, we have

$$P_S = \begin{cases} \sum_{j=0}^{L-1} \pi_j P_{S|j} & \text{infinite population} \\ \frac{\sum_{j=0}^{M-1}(M-j)\pi_j P_{S|j}}{\sum_{j=0}^{M-1}(M-j)\pi_j} & \text{finite transmitters} \end{cases} \tag{3.6}$$

If we count the number of successful packets transmitted in a long interval of time, and divide by the length of this interval, we will get a variable whose expected value is γ, the number of successful packets over time. We can find γ from $P_{S|j}$ as

$$\gamma = \begin{cases} \sum_{j=0}^{L-1} \lambda \pi_j P_{S|j} & \text{infinite population} \\ \sum_{j=0}^{\min(L,M)-1} \lambda \pi_j (M-j) P_{S|j} & \text{finite transmitters} \end{cases} \tag{3.7}$$

For the simple case of an infinite transmitter population with $L = 1$, which is a standard ALOHA channel, $\pi_0 = \exp(-\lambda/\mu)$ and $P_{S|0}$ is the probability that no arrivals occur before the packet is completed. Both the time to completion of the message and the time until the next arrival are exponentially distributed, so we can easily find

$$P_{S|0} = \frac{\mu}{\lambda + \mu} \tag{3.8}$$

$$P_S = \frac{e^{-g}}{1+g} \tag{3.9}$$

and

$$\frac{\gamma}{\mu} = \frac{ge^{-g}}{1+g} \tag{3.10}$$

This case was previously analyzed by Ferguson [FERG75] and by Bellini and Borgonovo [BELL80].

For the more difficult case of $L > 1$, we can find $P_{S|j}$ recursively as follows. A packet finding the system in state j will cause a transition to state $j+1$. From state $j+1$, we have the following probabilities for the next transition.

$$P_A(j) \triangleq \Pr(\text{next transition is due to an arrival}) = \frac{\lambda}{(j+1)\mu + \lambda} \tag{3.11}$$

$$\begin{aligned} P_B(j) &\triangleq \Pr(\text{next transition is due to another packet completing}) \\ &= \frac{j\mu}{(j+1)\mu + \lambda} \end{aligned} \tag{3.12}$$

$$\begin{aligned} P_C(j) &\triangleq \Pr(\text{next transition is due to the tagged packet completing}) \\ &= \frac{\mu}{(j+1)\mu + \lambda} \end{aligned} \tag{3.13}$$

Because of the memoryless property of the exponential packet length distribution, at every state transition, given that no errors have occurred and that the state entered is state $\hat{\jmath}$, the probability that the packet will be successful is the same as $P_{S|\hat{\jmath}-1}$. So we have the equations

$$\begin{aligned} P_{S|0} &= P_A(0)P_{S|1} + P_C(0) && j = 0 \\ P_{S|j} &= P_A(j)P_{S|j+1} + P_B(j)P_{S|j-1} + P_C(j) && 1 \le j \le L-1 \\ P_{S|j} &= 0 && j \ge L \end{aligned} \tag{3.14}$$

or

$$0 = -(\mu + \lambda)P_{S|0} + \lambda P_{S|1} + \mu$$

$$0 = -((j+1)\mu + \lambda)P_{S|j} + \lambda P_{S|j+1} + j\mu P_{S|j-1} + \mu \qquad 1 \leq j < L-1 \tag{3.15}$$

$$0 = -(L\mu + \lambda)P_{S|L-1} + (L-1)\mu P_{S|L-2} + \mu$$

We define $\underline{\boldsymbol{\Psi}}$ to be the vector such that $[\underline{\boldsymbol{\Psi}}]_j = P_{S|j+1}$. In matrix form,

$$\underline{\mathbf{0}} = \mathbf{R}\underline{\boldsymbol{\Psi}} + \mu\underline{\mathbf{1}} \tag{3.16}$$

where $\underline{\mathbf{0}}$ is the $L \times 1$ vector of zeros, $\underline{\mathbf{1}}$ is the $L \times 1$ vector of ones, and $\mathbf{R}$ is the matrix

$$\mathbf{R} = \begin{pmatrix} -(\mu+\lambda) & \lambda & 0 & 0 & \cdots & 0 \\ \mu & -(2\mu+\lambda) & \lambda & 0 & \cdots & 0 \\ 0 & 2\mu & (3\mu+\lambda) & \lambda & \cdots & 0 \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & (L-2)\mu & -((L-1)\mu+\lambda) & \lambda \\ 0 & \cdots & 0 & 0 & (L-1)\mu & -(L\mu+\lambda) \end{pmatrix} \tag{3.17}$$

A similar analysis for finite transmitters gives the matrix

$$\mathbf{R} = \begin{pmatrix} -\left({\mu+ \atop (M-1)\lambda}\right) & (M-1)\lambda & 0 & 0 & \cdots & 0 \\ \mu & -\left({\mu+ \atop (M-2)\lambda}\right) & (M-2)\lambda & 0 & \cdots & 0 \\ 0 & 2\mu & \left({3\mu+ \atop (M-3)\lambda}\right) & (M-3)\lambda & \cdots & 0 \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & (L-2)\mu & -\left({(L-1)\mu+ \atop (M-L+1)\lambda}\right) & (M-L+1)\lambda \\ 0 & \cdots & 0 & 0 & (L-1)\mu & -\left({L\mu+ \atop (M-L)\lambda}\right) \end{pmatrix} \tag{3.18}$$

We can solve Eqn. 3.16 for $\underline{\boldsymbol{\Psi}}$ as

$$\underline{\boldsymbol{\Psi}} = -\mu\mathbf{R}^{-1}\,\underline{\mathbf{1}} \tag{3.19}$$

Thus, having found $P_{S|j}$, we can find P_S and γ.

3.B Throughput Analysis

Because it is possible for a packet to be successfully transmitted even if overlapping transmissions occur, there are several steps required in finding the throughput. For an infinite population, throughput S is defined as

$$S = \sum_{j=0}^{\infty} j(\text{Fraction of time spent transmitting } j \text{ successful packets}) \qquad (3.20)$$

For finite transmitters, S_i, the throughput for user i is the fraction of time spent transmitting successful packets. Network throughput can be found as the sum of user throughputs, which is simply MS_i, since all users are identical.

Unfortunately, the probability of a packet being successful depends on its length as well as the state of the network upon its arrival. Thus, throughput is not simply γ/μ as it would be for fixed length packets (γ is the expected number of successful packets over time as in Section 3.A). Furthermore, for our definition of success, a packet cannot be counted as successful until it is completed, as an error anywhere will cause the entire packet to be unsuccessful. Thus, for determining success or failure, it is not sufficient to know only the state of the Markov chain at the beginning of a transmission. For this reason, we follow the approach used by Brazio and Tobagi [BRAZ84], and introduce an auxiliary Markov chain which traces the evolution of a tagged packet until either an error occurs or the packet transmission is successfully completed. We define S' to be the subset of the state space S consisting of all states from which a successful transmission can begin. An arrival to a state in S' will cause a transition to a state in the auxiliary Markov chain.

We define the following parameters:

$P_{S|j,\tau}$ is the probability that a packet is successful given that

i) it is of length τ,

ii) the radio is not busy when the packet arrives,

iii) it finds the network in state j upon arrival, and

iv) the protocol and channel state j allow it to be transmitted (in the channel load sense model discussed in Section 6).

T_s is the successful length of a packet, which is 0 if the packet is unsuccessful, and is the packet's length if it is successful.

$E(T_s|j)$ is the expected successful length of a packet given that the network is in state j upon its arrival, and that the packet was transmitted.

$P_{i_{IDLE}}(j)$ is the probability that user i is not transmitting given state j, as in Section 3.A.

$E(T_s|j)$ is found from $P_{S|j,\tau}$ by removing the condition on the packet length. Thus,

$$E(T_s|j) = \int_0^\infty P_{S|j,\tau}\, \tau f_T(\tau)\, d\tau \tag{3.21}$$

where $f_T(\tau)$ is the pdf of the packet lengths. Here, $f_T(\tau) = \mu e^{-\mu\tau}$. We note that $P_{S|j}$ derived in Section 3.A is

$$P_{S|j} = \int_0^\infty P_{S|j,\tau} f_T(\tau) d\tau \tag{3.22}$$

For an infinite population, throughput is given by

$$S = \sum_{j=0}^{\infty} \lambda \pi_j E(T_s|j) = \sum_{j=0}^{L-1} \lambda \pi_j E(T_s|j) \tag{3.23}$$

As shown in Appendix A, for finite transmitters, user throughput is given by

$$S_i = \sum_{j=0}^{M-1} \lambda \pi_j P_{i_{IDLE}}(j) E(T_s|j) \tag{3.24}$$

As in Section 3.A,

$$P_{i_{IDLE}}(j) = \frac{M-j}{M} \qquad \text{for } i = 1, 2, ..., M \tag{3.25}$$

So, network throughput S is

$$S = MS_i = \sum_{j=0}^{\min(L,M)-1} \lambda \pi_j (M-j) E(T_s|j) \tag{3.26}$$

3.C Calculation of $E(T_s|j)$

The simple case $L = 1$, infinite transmitter population, was analyzed by Ferguson [FERG75] and by Bellini and Borgonovo [BELL80]. The probability of a packet of length τ being correct, $P_{S|0,\tau}$, is the probability of no new arrivals in time τ, or

$$P_{S|0,\tau} = e^{-\lambda \tau} \tag{3.27}$$

From equation 3.21,

$$\begin{aligned} E(T_s|0) &= \int_0^\infty e^{-\lambda\tau}\, \tau \mu e^{-\mu\tau}\, d\tau \\ &= \frac{\mu}{\lambda+\mu} \int_0^\infty (\lambda+\mu)\tau e^{-(\lambda+\mu)\tau}\, d\tau \\ &= \frac{\mu}{(\lambda+\mu)^2} = \frac{1/\mu}{(1+g)^2} \end{aligned} \tag{3.28}$$

which gives

$$S = \lambda e^{-g} \frac{1/\mu}{(1+g)^2} = \frac{g e^{-g}}{(1+g)^2} \tag{3.29}$$

The maximum throughput of 0.137 occurs at $g = \sqrt{2} - 1$.

To solve for $E(T_s|j)$ for $L > 1$, we consider the evolution of a packet that arrives and is transmitted. If $j \geq L$, $E(T_s|j) = 0$. If $j < L$, the packet will begin to be transmitted

with a non-zero probability of success. As the packet transmission progresses, the network state changes. Eventually, either an error occurs, in which case the transmission fails, or the packet is completed, and the transmission succeeds. This evolution is modeled by an auxiliary Markov chain with a state space S_{aux}, which is constructed from S as follows [BRAZ84].

First, we delete all states for which the probability of successful transmission is zero. For an infinite population, this leaves states $\{0, 1, ..., L\}$, and for finite transmitters, the states $\{0, 1, ..., \min(M, L)\}$. If $M \leq L$, all packets will be successful, so $E(T_s|j) = 1/\mu$, the average packet length. Thus, we will consider only cases where $L < M$. Next, we delete all states corresponding to no transmissions occurring, here the state $j = 0$. Finally, we add two absorbing states, Success and Failure. We reduce by μ the rate of all transitions from states j to states $j - 1$, $2 \leq j \leq L$, and add transitions from every non-absorbing state to the state Success at rate μ. A transition from the state L to the state Failure is added corresponding to the rate at which new transmissions are scheduled when in the state L, which is λ for an infinite population, and is $(M - L)\lambda$ for finite transmitters. Thus, we have the auxiliary Markov chains given in Figs. 3.3 and 3.4. If we index the states Success and Failure as the last two states, the transition rate matrix $\mathbf{R}^*$ is

$$\mathbf{R}^* = \begin{pmatrix} \mathbf{R} & \mu\underline{\mathbf{1}} & \varphi \\ \underline{\mathbf{0}} & 0 & 0 \\ \underline{\mathbf{0}} & 0 & 0 \end{pmatrix} \tag{3.30}$$

where $\mathbf{R}$ is the $L \times L$ sub-matrix corresponding to transitions between the states $\{1, 2, ..., L\}$ of the auxiliary Markov chain, φ is the $1 \times L$ sub-matrix corresponding to transitions to the state Failure, $\underline{\mathbf{1}}$ is the $1 \times L$ vector of ones, and $\underline{\mathbf{0}}$ is the $L \times 1$ vector of zeros. For infinite and for finite transmitters, $\mathbf{R}$ is identical to the corresponding matrix given in Section 3.A. Indeed, as we show in Appendix B, $P_{S|j}$ is simply the probability that the auxiliary Markov chain is in state success at time $\tau = \infty$ given that it was in state $j + 1$ at time $\tau = 0$, which can be found from $\underline{\Psi} = -\mu\mathbf{R}^{-1}\underline{\mathbf{1}}$ as shown earlier. Furthermore, the quantity

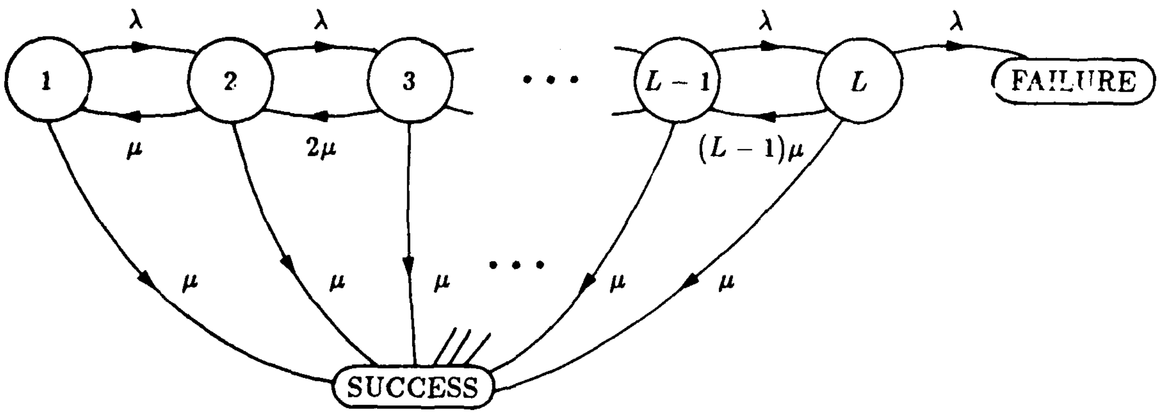

Figure 3.3. Auxiliary Markov chain for the infinite population model.

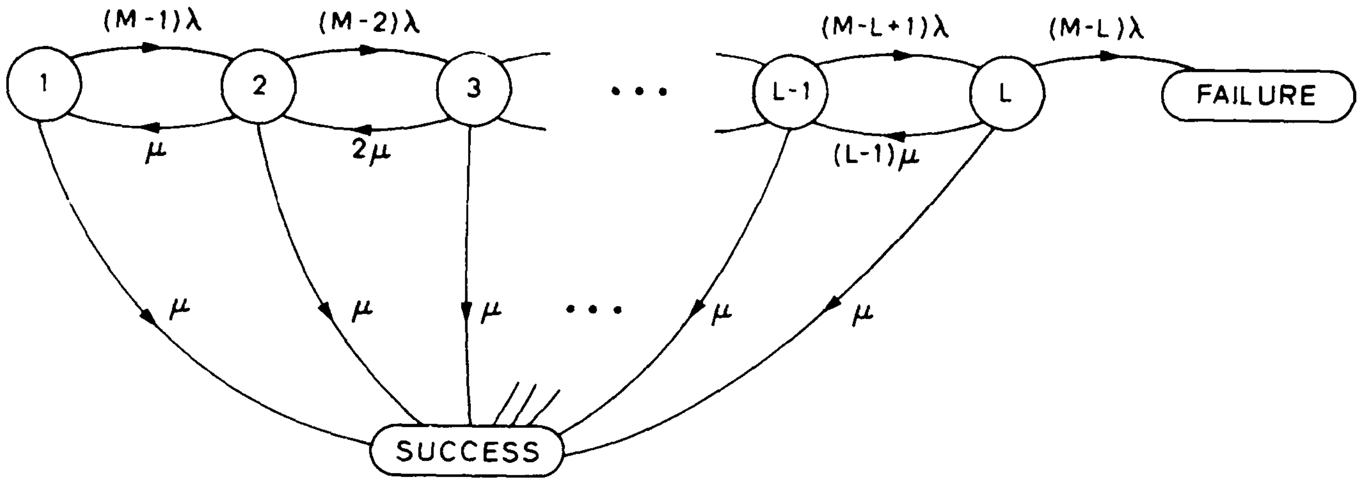

Figure 3.4. Auxiliary Markov chain for the finite transmitters model.

$E(T_s|j)$ is the expected time to reach state Success given that the auxiliary Markov chain starts in state j at time $\tau = 0$. This has been shown [BRAZ84] to be

$$E(T_s|j) = [\mu \mathbf{R}^{-2} \underline{\mathbf{1}}]_{j+1} \qquad j \in S' \tag{3.31}$$

The derivation is given in Appendix B.

3.D Packet Header Synchronisation Probability

Because this model does not specifically account for the receivers, packet header synchronization affects the probability of a packet being successful but it does not affect the state space. We define α_j to be the probability that the receiver captures the packet given j interfering transmissions. The contribution to throughput by packets that found the network in state j upon arrival will be reduced by the factor α_j. Similarly, γ and P_S will be reduced. For an infinite population, we have

$$P_S = \sum_{j=0}^{L-1} \pi_j \alpha_j P_{S|j} \tag{3.32}$$

$$\gamma = \sum_{j=0}^{L-1} \lambda \pi_j \alpha_j P_{S|j} \tag{3.33}$$

$$S = \sum_{j=0}^{L-1} \lambda \pi_j \alpha_j E(T_s|j) \tag{3.34}$$

and for finite transmitters, we have

$$P_S = \frac{\sum_{j=0}^{M-1} (M-j) \pi_j \alpha_j P_{S|j}}{\sum_{j=0}^{M-1} (M-j) \pi_j} \tag{3.35}$$

$$\gamma = \sum_{j=0}^{M-1} \lambda \pi_j (M-j) \alpha_j P_{S|j} \tag{3.36}$$

$$S = \sum_{j=0}^{M-1} \lambda \pi_j (M-j) \alpha_j E(T_s|j) \tag{3.37}$$

Again, if $L < M$, $P_{S|j}$ and $E(T_s|J)$ are zero for $j \geq L$.

3.E Results

In Figs. 3.5 and 3.6 we plot the throughput S, γ/μ, and P_S for $L = 10$ for an infinite population, and for finite transmitters ($M = 25$). These results are similar to those found by Pursley [PURS83]. We notice that when P_S drops below one, γ/μ becomes greater than the throughput, but that the curves are similar.

In Fig. 3.7, we plot normalized throughput S/L vs. normalized offered traffic g/L for several values of L for the infinite population model. Although L is limited to whole numbers, a continuous function is shown in the following plots to make them easier to read. The results are analogous to those found by Musser and Daigle [MUSS82] for fixed size packets. We notice that as L increases, the maximum throughput comes closer to the capacity L, but performance is more sensitive to the offered traffic rate.

4. Realistic Channel Model

The performance of spread spectrum radio channels using error correction coding in a CDMA environment has been analyzed by many authors. The models derived either bound or approximate the channel performance. Several difficulties arise in incorporating these models into the Markovian network model described earlier. First, the detailed channel models require a large amount of computation for even moderate numbers of interfering users. Thus, numerical results are not tractable over the entire range of interest. Second, the performance of decoders is often modeled using somewhat loose bounds. These bounds are poor approximations to actual performance for regions of high BER. However, these high BER regions are within the range of interest for network performance. Finally, the use of decoders introduces memory into the system, as the output bit depends on the received signal over an interval extending many bits into the past. In this section, we discuss the various approximations used to overcome these difficulties and verify the validity of the approximations. We then present numerical results for several cases.

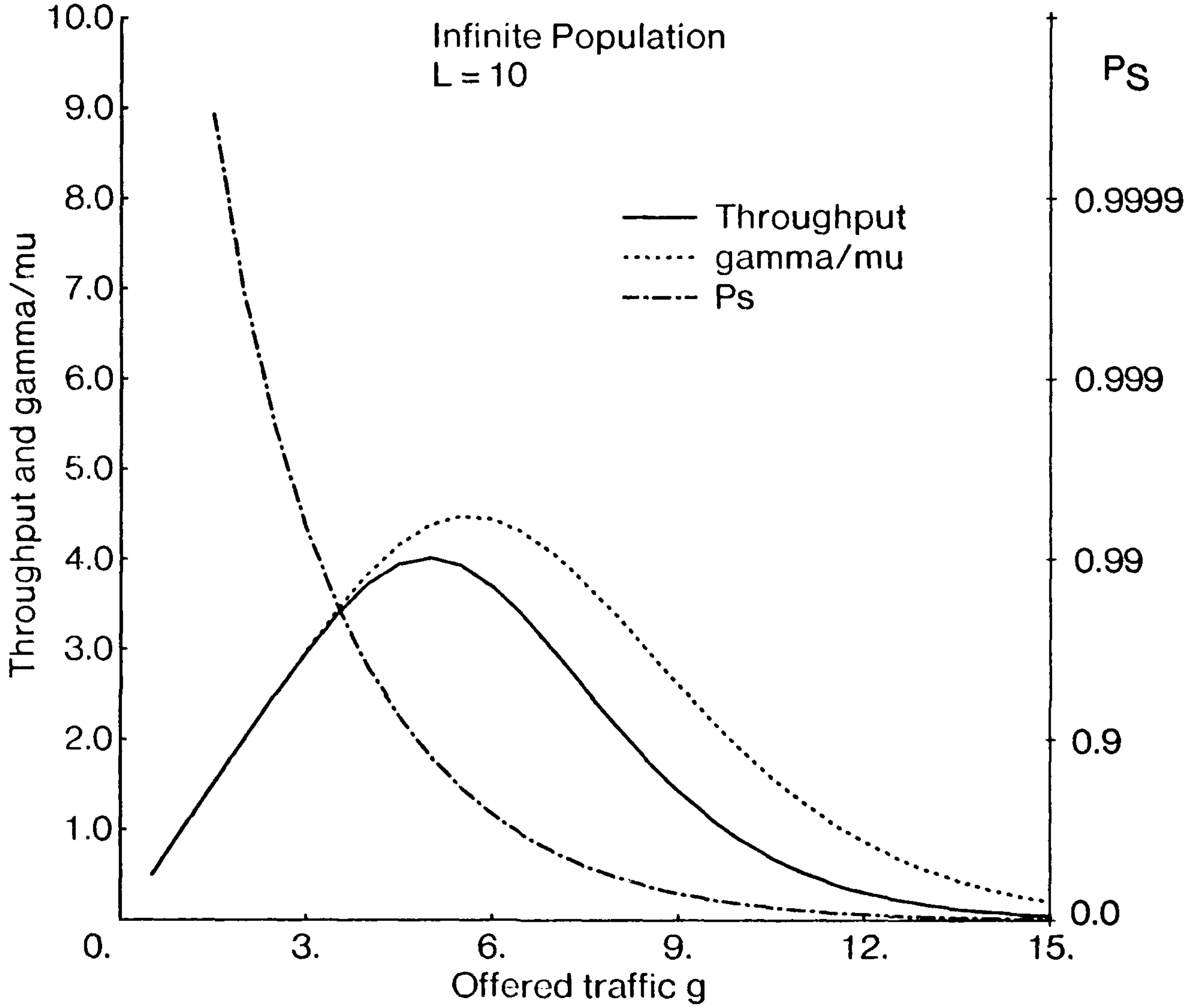

Figure 3.5. Throughput, γ/μ, and P_S versus offered traffic g for infinite population and $L = 10$.

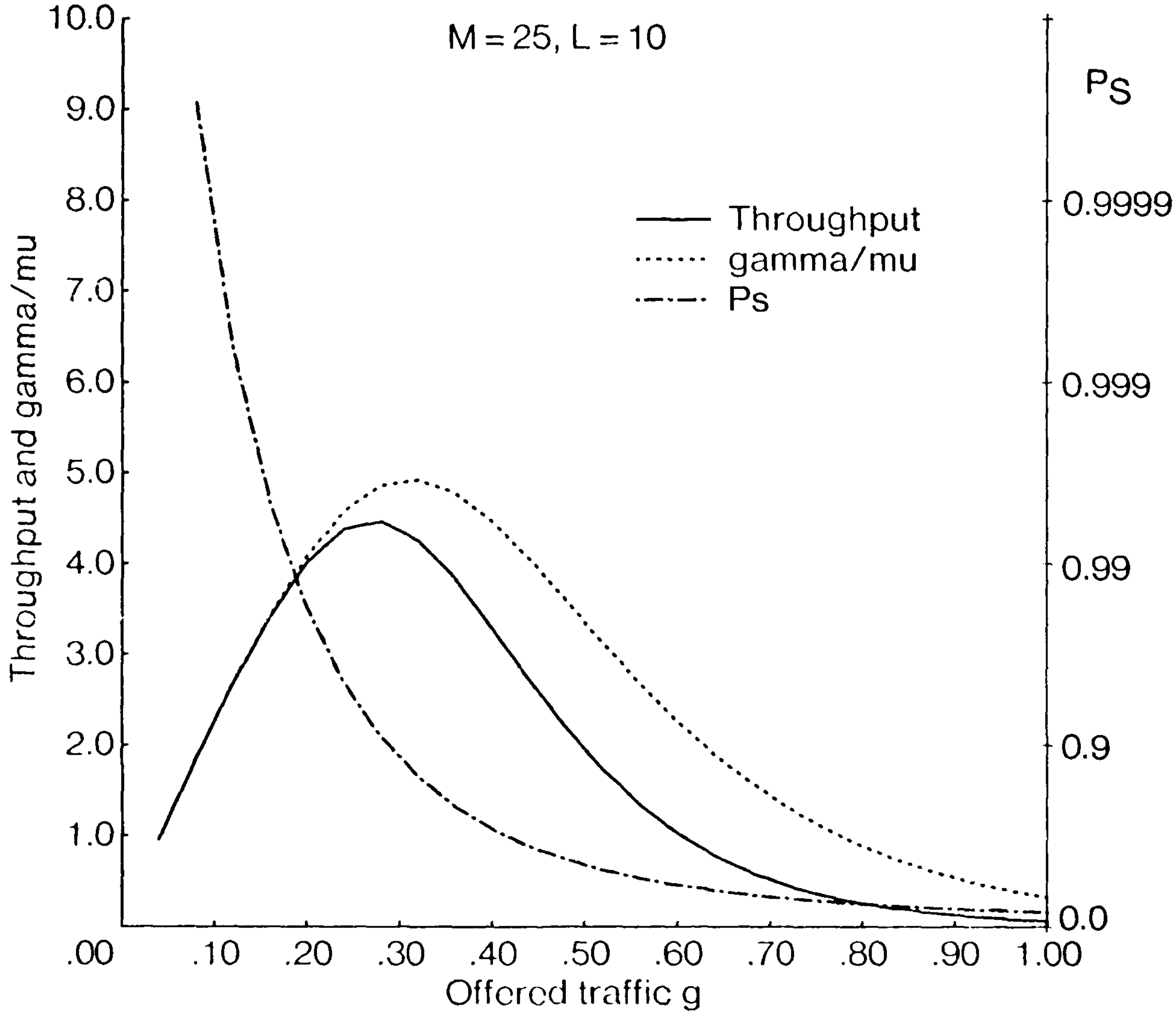

Figure 3.6. Throughput, γ/μ, and P_S versus offered traffic g for finite transmitters ($M = 25$) and $L = 10$.

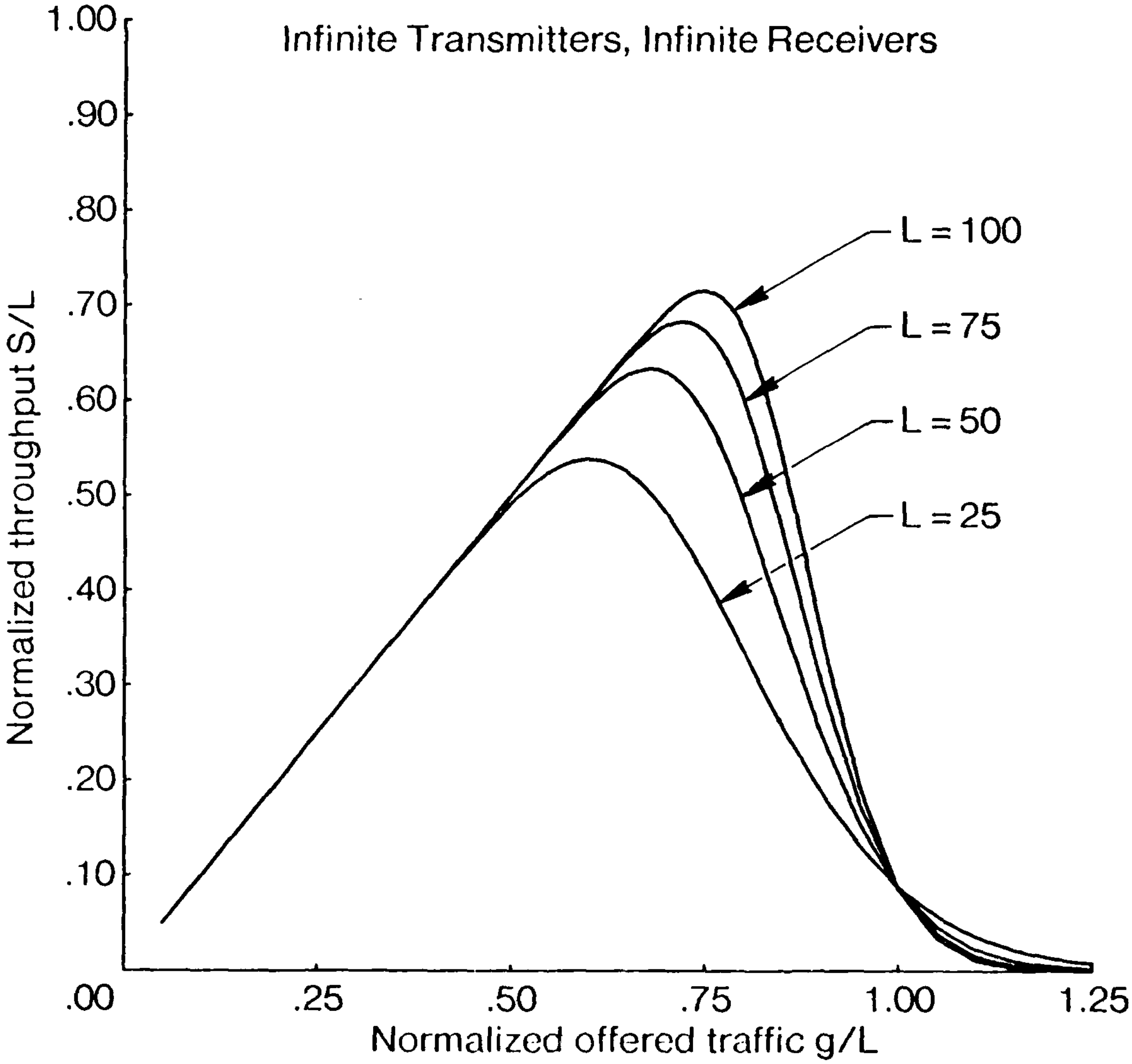

Figure 3.7. Normalized throughput S/L versus normalized offered traffic g/L for an infinite population.

4.A Radio Channel Model [TAIP84]

As mentioned previously, a large number of detailed analyses of DS-CDMA channels have been published. We choose to incorporate a model which is numerically tractable even for large numbers of interfering users. The model was developed and analyzed by Taiple [TAIP84]. It relies on several worst case assumptions, and does not require specific characteristics of the set of codewords used for the PN sequence. Therefore, some of the approximations used are lower bounds which are believed to be only moderately tight over some of the range of interest.

The channel model is of a coherent BPSK DS-CDMA communications system. It does not account for fading, multipath, or jamming. The received signal power is identical for all transmitter-receiver pairs, and the thermal noise level is identical for all users, so all receptions have the same received bit energy to noise energy E_b/N_0. The receiver is assumed to be perfectly synchronized with the transmitted signal, which includes carrier phase, pseudo-noise code chip timing, and bit timing. In addition, a worst case is assumed for the interfering signals; all interfering signals are in phase and chip aligned with the desired signal.

We adopt the following terminology. Bits refer to the uncoded information stream generated by the user. Symbols refer to the output of the error correction encoder. Thus, for a rate 1/2 code, there will be 2 symbols per bit. Chips refer to the pseudo-noise coded information.

The general channel model is as follows. Information to be transmitted is formed into packets of data bits. FEC coding is applied to these bits, resulting in binary symbols. Each symbol is multiplied by a pseudo-noise (PN) code, resulting in a number of binary chips. We denote the number of chips per data bit as N. N is the total bandwidth expansion due to both FEC coding and PN spreading. Finally, the chips are input to a modulator which phase modulates an RF carrier. Thermal noise and a number of interfering signals

are added to the transmitted carrier, resulting in a corrupted received signal. The receiver coherently demodulates the signal and despreads it by multiplying by the PN code stream. It then integrates over one symbol, and makes a hard decision on the symbol value. We denote the symbol time by T. The decoder then estimates the source's data bit stream from the corrupted symbol stream received. If any bit is incorrectly estimated, we declare the entire packet in error. We again implicitly assume additional error detection, such as a CRC code, but ignore the overhead required. The radio and channel model is diagrammed in Fig. 4.1.

The mathematical model is a special case of the channel model given by Pursley [PURS77]. All users are identical, so for notational convenience, we assume that the source is user 1 and the destination is user 0. In the general model, the received signal $r(t)$ is given by

$$r(t) = n(t) + \sum_{k=1}^{J} \sqrt{2P} a_k(t - \tau_k) b_k(t - \tau_k) \cos(\omega_c t + \phi_k) \tag{4.1}$$

where J is the number of signals simultaneously arriving at user 0 at time t, a_k is the pseudo-noise code stream of transmitter k, b_k is the FEC coded symbol stream of transmitter k, ϕ_k is the phase shift from transmitter k to user 0, τ_k is the time delay from transmitter k to user 0, P is the received signal power, and $n(t)$ is the additive white Gaussian noise (WGN).

In the specific channel model examined, we assume that the symbols b_k and the code chips a_k are independent random variables assuming the values $+1$ and -1 with probabilities 1/2 and 1/2. In addition, because of the worst case assumption, τ_k is a multiple of the chip time T/N and ϕ_k is a multiple of 2π.

Because of the symmetry of the channel, the probability of error is the same whether the transmitted symbol is a $+1$ or a -1. Thus, we will only analyze the case when the

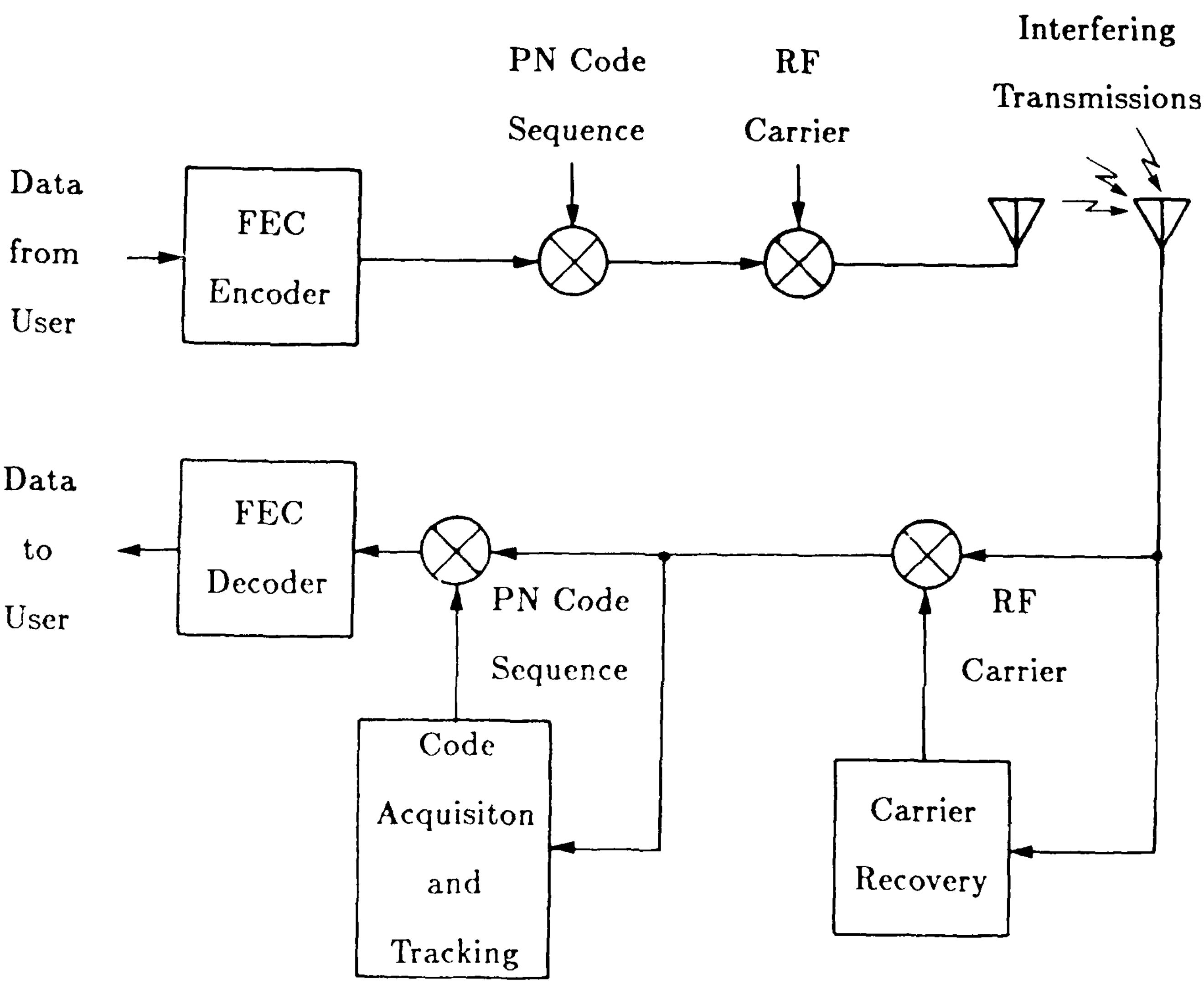

Figure 4.1. Spread spectrum radio model.

transmitted symbol is a +1. We define $c_k(t)$ as the product of the received chip from user k with the code stream of user 1, so

$$c_k(t) = a_k(t - \tau_k) b_k(t - \tau_k) a_1(t - \tau_k)$$

Because of the assumption of chip alignment, we can unambiguously define the discrete chips

$$c_{k,l} = c_k\left(t = \frac{(l - 1/2)T}{N}\right) \qquad l = 1 \text{ to } N$$

The random variables $\{c_{k,l}\}$ are jointly independent Bernoulli trials.

The output of the correlation receiver is

$$Z = \sqrt{\frac{2}{T}} \int_0^T r(t) a_k(t) \cos(\omega_c t) dt \tag{4.2}$$

This reduces to

$$Z = \sqrt{E_s} \left(1 + \sum_{k=2}^{J} \sum_{l=1}^{N} c_{k,l}\right) + \mathcal{N} \tag{4.3}$$

where $E_s = PT$ is the energy per symbol, and $\mathcal{N}$ is a Gaussian random variable with variance $N_0/2$. It can be seen that $Z = \mathcal{D} + \mathcal{I} + \mathcal{N}$, where $\mathcal{D}$ is the desired symbol, $\mathcal{I}$ is the sum of the interfering signals, and $\mathcal{N}$ is thermal noise. A symbol error occurs if $Z \leq 0$. Knowing $\mathcal{D} + \mathcal{I}$, we can find $\Pr(Z \leq 0)$ from standard communications theory. But $\mathcal{D}$ is known to be $\sqrt{E_s}$ and $\mathcal{I}$ has a known distribution, so we can find the mean probability of symbol error given J simultaneous transmissions, $\overline{P}_{e,symbol}(J)$, by averaging over all values of $\mathcal{I}$.

$$\Pr\left(\mathcal{I} = \frac{\sqrt{E_s}(2i - (J-1)N)}{N}\right) = \binom{(J-1)N}{i} \qquad 0 \leq i \leq (J-1)N \tag{4.4}$$

$$\overline{P}_{e,symbol}(J) = \sum_{i=0}^{(J-1)N} \binom{(J-1)N}{i} 2^{-(J-1)N} \, (i) \tag{4.5}$$

where

$$p(i) = Q\left(\sqrt{2E_s/N_0}\left(1 + \frac{2i - (J-1)N}{N}\right)\right) \tag{4.6}$$

and $Q(x)$ is the Marcum Q function,

$$Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^\infty e^{-\gamma^2/2}d\gamma \tag{4.7}$$

4.B Error Correction Coding Model

The performance of CDMA radio networks can often be improved by introducing FEC coding. We analyze the network performance using a rate 1/2 constraint length 7 convolutional code with Viterbi decoding. The decoder performance analysis is discussed in some detail in Appendix C. In this section, we will briefly discuss the decoder model and several limitations. From the analysis of the decoder, we can find the so-called probability of first error, P_E, which is the probability of a bit being in error given that no earlier bits were in error. Since we declare an entire packet to be in error if any bit is in error, this is the appropriate performance measure. Fig. 4.2 shows P_E vs. the input symbol error rate.

Taiple [TAIP84] shows that for fixed length packets of length $\mathcal{L}$ in a slotted system, the probability of a packet being correct is lower bounded by $(1 - P_E)^{\mathcal{L}}$. For packets of 1000 bits, if P_E is as high as 10^{-3}, there is still at least a 37% chance that the packet will be correct. For $P_E = 5 \times 10^{-3}$, this drops to at least 0.67%. This indicates that we should strive for a decoder model which is accurate up to P_E of at least 5×10^{-3}. Unfortunately, as discussed in Appendix C, the analysis uses a union bound which becomes very loose for P_E above about 10^{-3}. The rapid increase in P_E in the range 10^{-3} to 10^{-2} can be seen in Fig. 4.2. Thus, the decoder model gives pessimistic results for P_E in this range.

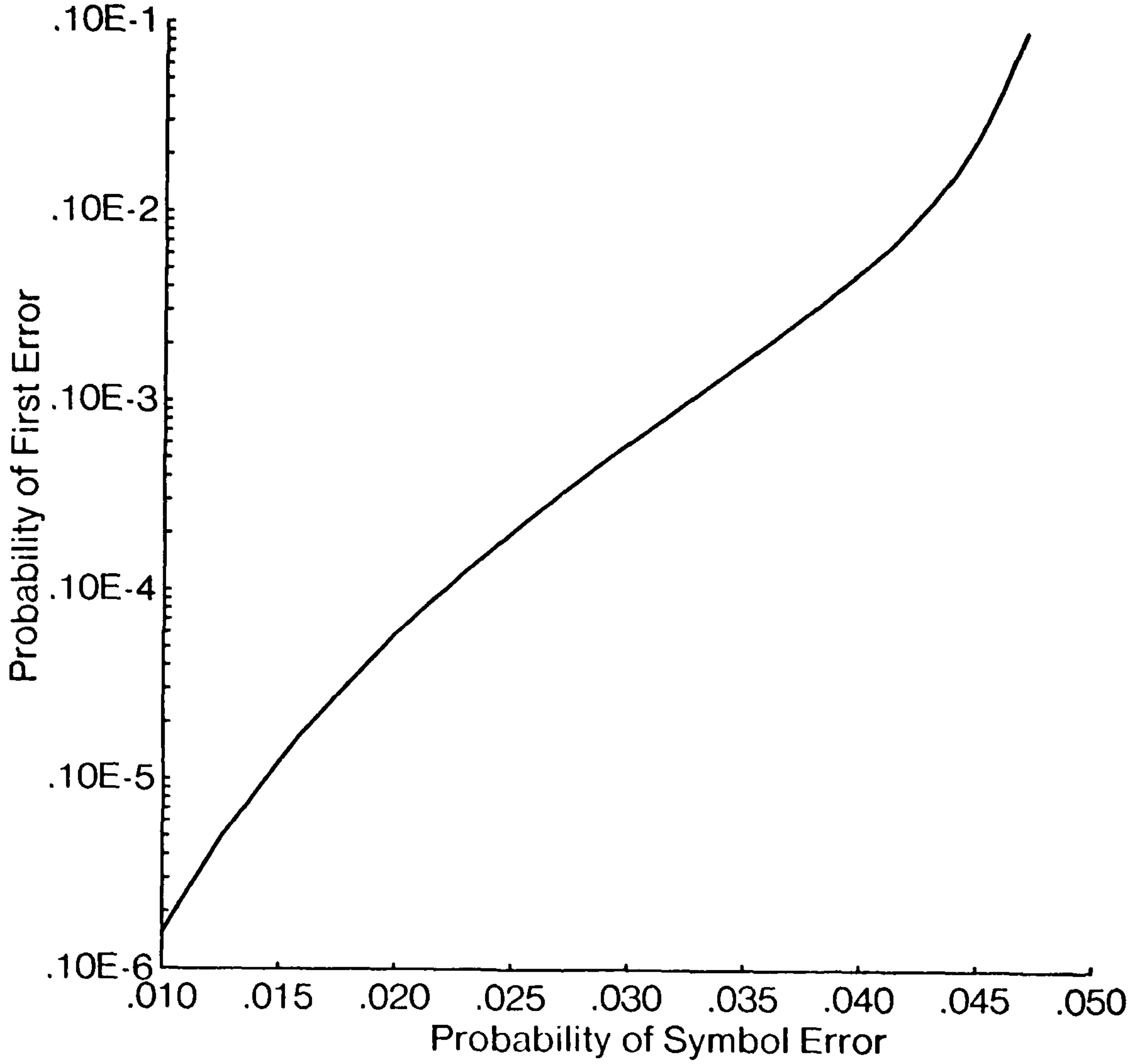

Figure 4.2. Probability of first error versus symbol error rate.

4.C Refined Network Model

From the channel model and the FEC decoder model, we can find $P_E(j)$, the probability of first error given a symbol error rate of $\overline{P}_{e,symbol}(j)$. $P_E(j)$ vs. j is plotted in Fig. 4.3 for $E_b/N_0 = 8.0$ for several values of N.

As shown in Appendix D, for a packet which does not visit the state $L+1$ and spends $\mathcal{L}_j$ bits in state j, the probability of correct reception is approximately bounded by

$$P_C \gtrsim \prod_{j=1}^{L} (1 - P_E(j))^{\mathcal{L}_j} \tag{4.8}$$

We can account for this in the network model by introducing a time varying Poisson process that generates errors at a rate ϵ_j while the tagged packet is in state j. Specifically, this is accomplished by adding transitions from each non-absorbing state j in the auxiliary Markov chain to the state Failure at at rate ϵ_j, as shown in Fig. 4.4. The probability that no errors occur while in state j for time τ_j is simply $e^{-\tau_j \epsilon_j}$. For a small bit time Δt, $\tau_j \approx \mathcal{L}_j \Delta t$ for some number of bits $\mathcal{L}_j$. The probability that a packet is successful is the product of the probabilities that no errors occur during any state j.

$$P_C = \prod_{j=1}^{L} e^{-\tau_j \epsilon_j} \approx \prod_{j=1}^{L} (e^{-\Delta t \epsilon_j})^{\mathcal{L}_j} \tag{4.9}$$

If we set $1 - P_E(j) = e^{-\Delta t \epsilon_j}$, the probability of a packet being correct will be precisely the bound given by Eqn. 4.8. Thus,

$$\epsilon_j = -\frac{1}{\Delta t} \ln(1 - P_E(j)) \quad \text{or} \quad \frac{\epsilon_j}{\mu} = -\frac{\ln(1 - P_E(j))}{\mu \Delta t} = -b \ln(1 - P_E(j)), \tag{4.10}$$

here $b = 1/(\mu \Delta t)$ is the average number of bits per packet.

Several approximations are implied by this model.

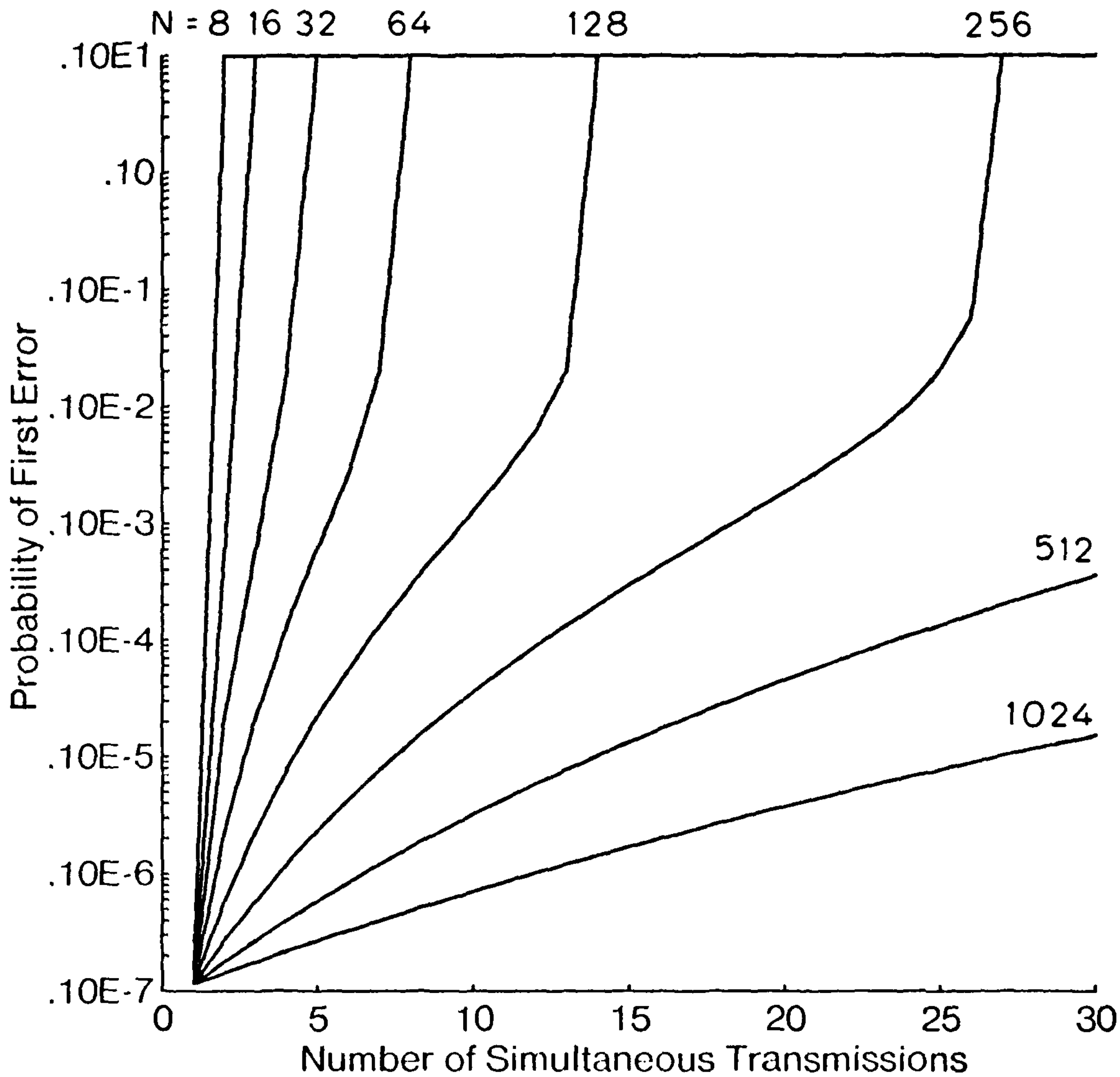

Figure 4.3. Probability of first error versus number of simultaneous transmissions.

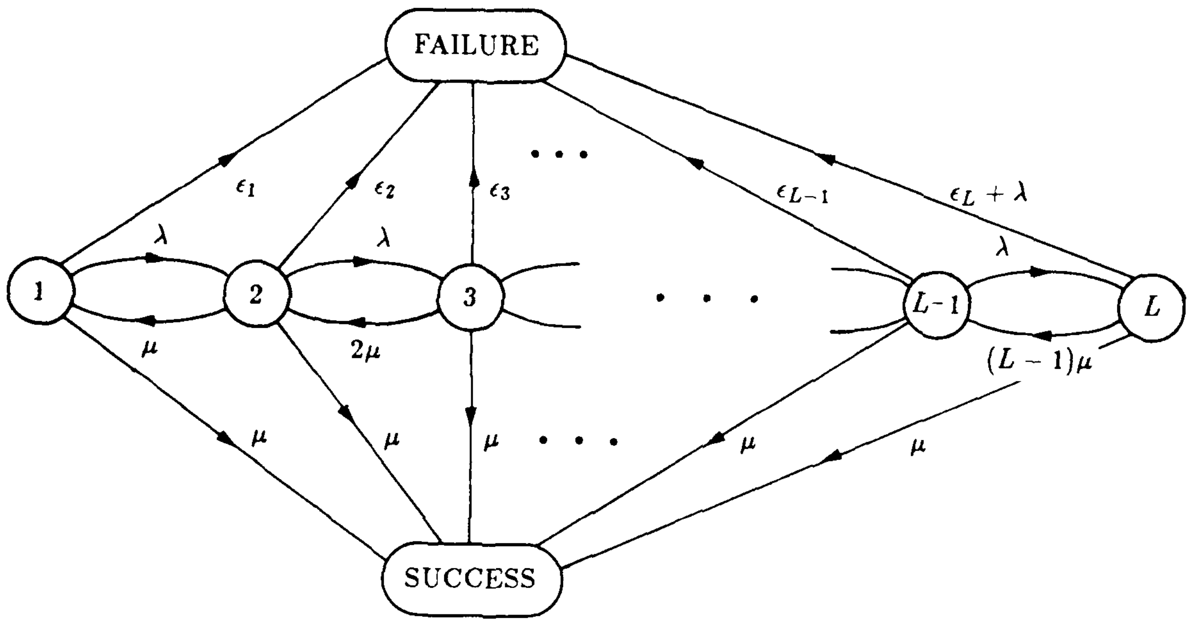

Figure 4.4. Auxiliary Markov chain for modeling a general $P_E(j)$.

i) First, the channel and decoder models yield an error process which occurs at bit boundaries. There are two approaches to incorporating this into the network model. One approach is to change the network model to a discrete time model, with arrivals and departures occurring at bit boundaries. For bit times which are short in comparison to the average holding times in the states, this model will be very similar to the continuous time model, except that there will now be transitions occurring between non-neighboring states. These transitions will have very low probabilities for short bit times, as the arrival or departure rate per bit is low. We chose the alternative approach, which is to approximate the error process as a continuous time process, in which errors are generated by a time varying Poisson process. This yields a simpler network model and will give accurate results for average packet lengths of 1000 bits.

ii) Second, the network model requires a cut-off L such that errors occur with probability one for $j > L$. No such cut-off exists in the physical channel, since the probability of bit error is never greater than 1/2, so there is always a non-zero probability of correct reception. Introducing the limit L underestimates the throughput, as some successful packets will be counted as unsuccessful. Nevertheless, if L is chosen such that $(1 - P_E(L+1))^H$ is small for H equal to the mean holding time in state $L+1$, the probability of a packet successfully surviving a visit to state $L+1$ will be small.

The choice of L is simple in the case of the coded channel because of the decoder model used. As explained in Appendix C, the bound for the decoder model is not known to be accurate for P_E above 10^{-2}. For this reason, the model effectively truncates the performance in this range, and sets $P_E = 1$ when the bound indicates $P_E > 10^{-2}$. Thus, L is chosen to be the largest j such that $P_E(j) \leq 10^{-2}$.

iii) Another approximation implied by the model is that the addition of an error process does not affect the existing transition rates for the auxiliary Markov chain. In reality, the existing transition rates leaving state j will be reduced by $1 - P_E(j)$ per bit. However, for

$P_E(j)$ as high as 10^{-2}, this will only be a 1% reduction. Thus, this approximation has a negligible effect on throughput for the coded channel model.

iv) The final assumption made in incorporating a realistic channel model into the network model is that the channel behavior is memoryless. It can be shown that for some of the cases considered, the decoder performance depends on past states most of the time. Nevertheless, the approximate bounds described below show that the variation in throughput due to this assumption is small. Thus, even though the model does not exactly parallel the physical system, the results closely match the ideal performance of the system.

The decoded memory typically extends 30 bits into the past [CLAR81]. Thus, P_E at time τ depends not only upon the current state, but also upon the states visited during the last 30 bits, or 60 symbols. The arrival and departure processes are both Poisson, so the holding time in a state j is exponentially distributed, with rate $\lambda + j\mu$ for the infinite population model, or $(M-j)\lambda + j\mu$ for the finite transmitters model. The mean holding time in state j is thus

$$\begin{aligned} \frac{1}{\lambda + j\mu} &= \frac{1/\mu}{g+j} && \text{infinite population} \\ \frac{1}{(M-j)\lambda + j\mu} &= \frac{1/\mu}{(M-j)g+j} && \text{finite transmitters} \end{aligned} \tag{4.11}$$

For $1/\mu = 1000$ bits, the average holding time is

$$\frac{1000}{g+j} \qquad \text{or} \qquad \frac{1000}{(M-j)g+j}$$

For the coded channel, with $E_b/N_o = 8.0$, $N = 64$ chips per bit, the cut-off $L = 7$, and the maximizing values of offered traffic g are 3.01 (infinite population), 0.04 (80 transmitters), and 0.53 (10 transmitters). It can be seen that even for the state $j = L$, the average holding times are $1000/(7+3.01)$, $1000/(7+80\cdot 0.04)$, and $1000/(7+10\cdot 0.53)$, all of which are about three times the decoder memory of 30 bits. In these cases, we can ignore the

transient behavior which occurs during the 30 bits following a state change. Unfortunately, for larger N, the mean holding time in some states is as low as 10 or 20 bits. For these cases, the decoder output is very likely to depend upon the previous state for a large fraction of the time spent in these states.

Nevertheless, even for these few extreme cases, the probability that the number of interfering transmissions j varied widely during the last 30 bits is not large. For offered traffic g in the range which gives maximum throughput, most of the steady state probability is concentrated in the states with intermediate values of $P_E(j)$. We find that $P_E(j)$ varies slowly with j over these intermediate ranges (see Fig. 4.3). This is especially true for very large numbers of chips per bit, which are the cases where the memory may be significant. Thus, for those cases were the decoder memory cannot be ignored as a second order transient effect, the difference between $P_E(j)$, $P_E(j+1)$ and $P_E(j-1)$, and even $P_E(j+2)$ and $P_E(j-2)$ is small.

To verify this approximation, we calculated approximate bounds by using first $P_E(j+3)$ and then $P_E(j-3)$ as the error rates from state j. This was done for the infinite population model, with $E_b/N_0 = 8.0$ and with 256 chips per bit. For this case, the cut-off L is 26. For offered traffic g of 26.0, the transitions from j to $j+1$ occur at a rate $g\mu = 26.0/1000$ per bit, and the transitions from j to $j-1$ occur at a rate $j\mu$, which is upper bounded by $L\mu = 26/1000$ per bit. The probability of going from state j to a state $j' > j + \delta$ in 30 bits is less than the probability of more than δ arrivals from the Poisson arrival process (which is rate 0.026 per bit) in 30 bits, which is simply

$$1 - \sum_{i=0}^{\delta} \frac{e^{-0.78}(0.78)^i}{i!} \tag{4.12}$$

For $\delta = 3$, this comes to 0.0083. This probability is even smaller for offered traffic g less than 26.0. Similarly, the probability of going from state j to a state $j' < j - \delta$ in 30 bits is less than 0.0083. Thus, when the current network state is j, with probability greater

than 99%, the minimum symbol error rate encountered by any of the 60 symbols in the decoder memory will be no less than $P_E(j-3)$ and the maximum symbol error rate will be no greater than $P_E(j+3)$. Thus, substituting $P_E(j \pm 3)$ for the symbol error rate of state j will give approximate bounds to the performance of a decoder with a memory of 30 bits.

These bounds are fairly loose, since for many states, the variation will be less than ± 2 or ± 1 states with a very high probability. Also, even when the states $j+3$ or $j-3$ are visited, many of the last 60 symbols will have symbol error probabilities closer to $P_{e,symbol}(j)$ than $P_{e,symbol}(j \pm 3)$. Even so, as can be seen in Fig. 4.5, the maximum values of the throughput for the models which use $P_E(j \pm 3)$ are within 20% of the maximum throughput for the model which uses $P_E(j)$. Therefore, the approximate model which ignores the memory of the decoder yields meaningful results even for extreme cases where holding times are on the order of 30 bits.

4.D Standard ALOHA Channel

For comparison, we derive the throughput of the uncoded, unspread radio channel with thermal noise. In this case, $N = 1$ chip per bit, and also $L = 1$. The analysis is similar to the case with no thermal noise, which was presented in Section 3.C. However, in addition to new arrivals, there are also failures due to an error arriving before the packet is completed. Thus,

$$P_{S|0,\tau} = e^{-(\lambda+\epsilon_1)\tau} \tag{4.13}$$

$$E(T_s|0) = \frac{1/\mu}{(1+g+\epsilon_1/\mu)^2} \tag{4.14}$$

$$S = \frac{ge^{-g}}{(1+g)^2} \tag{4.15}$$

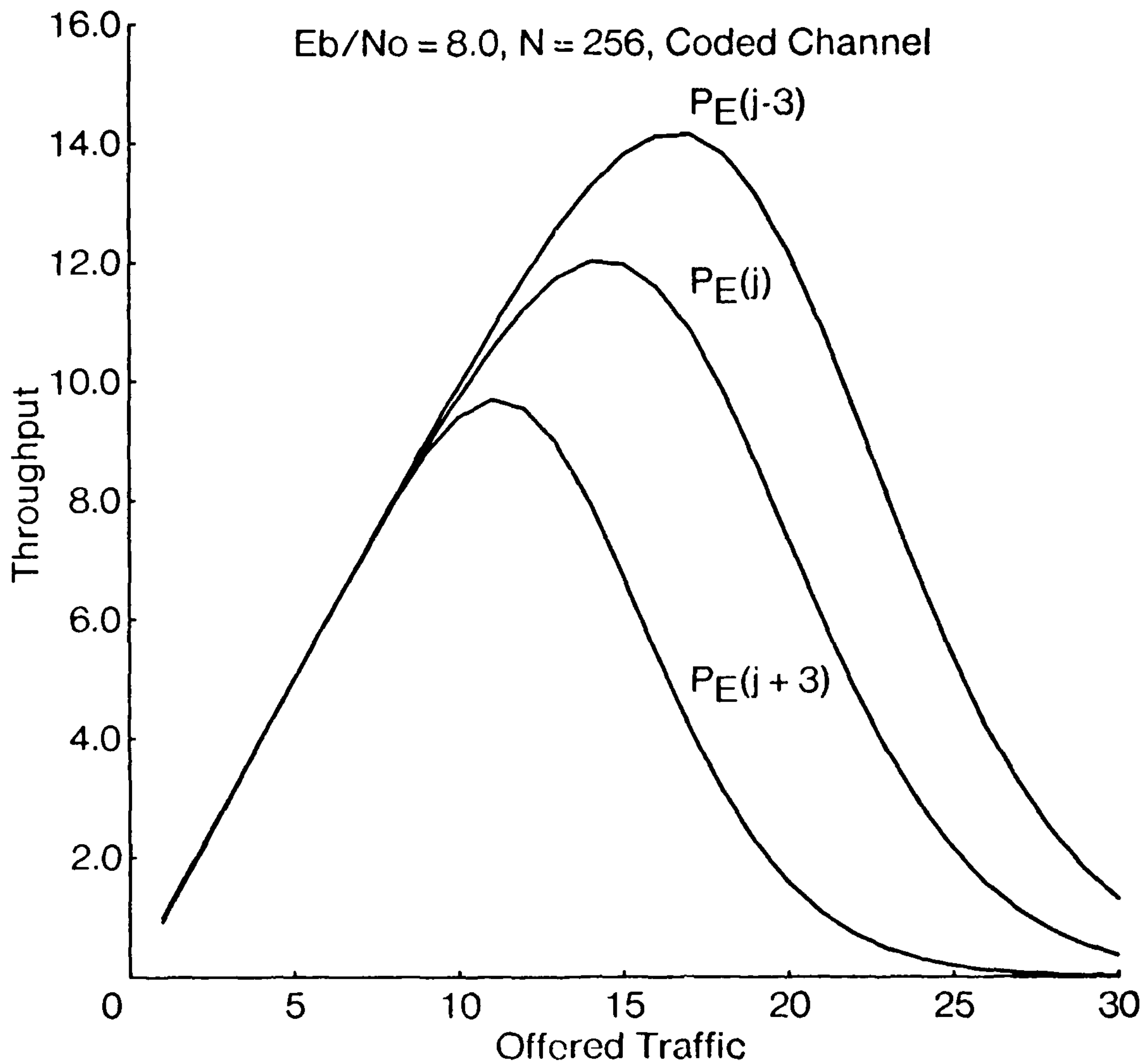

Figure 4.5. Throughput for bounds on the probability of symbol error.

For an uncoded channel with $E_b/N_o = 8.0$, 1000 bits per average packet, $\epsilon_1/\mu = 1.15\times10^{-5}$, which is much less than 1, so the extra term is negligible. Thus, the thermal noise has little effect on throughput, so the maximum throughput is again 0.137, achieved at $g = \sqrt{2} - 1$.

4.E Results

For a fair comparison of throughput, we state most results in terms of S/N, which is throughput divided by N, the number of chips per information bit. S/N indicates the throughput per unit bandwidth, where a bandwidth of one is required by an uncoded unspread signal.

Fig. 4.6 shows the normalized throughput S/N vs. the normalized offered traffic g/N, for the coded channel with $E_b/N_0 = 8.0$, for the infinite population model. Results are given for several values of N. This has a shape similar to Fig. 3.7.

The following curves show the effect of various parameters on the maximum throughput $S_{\max}$, achieved by maximizing over offered traffic g. For convenience, we refer to this maximum as throughput or normalized throughput, without specifically indicating the maximization over g.

Fig. 4.7 shows the throughput vs. N for several values of E_b/N_0, for the infinite population model. The corresponding values of S/N are plotted in Fig. 4.8. It can be seen that the performance becomes limited by multi-user interference for E_b/N_0 greater than about 20. Also, the normalized throughput increases monotonically with N. The maximum normalized throughput for the uncoded ALOHA channel ($N = 1$) is 0.137. Thus, over all of the ranges of E_b/N_0 and N examined, the normalized capacity of the spread spectrum channel with FEC coding is less than that of a non-spread channel with no coding.

We plot S/N vs. E_b/N_0 for the coded and the uncoded channels in Fig. 4.9, for 256 chips per bit. The improvement due to FEC coding is significant. Again, we notice that

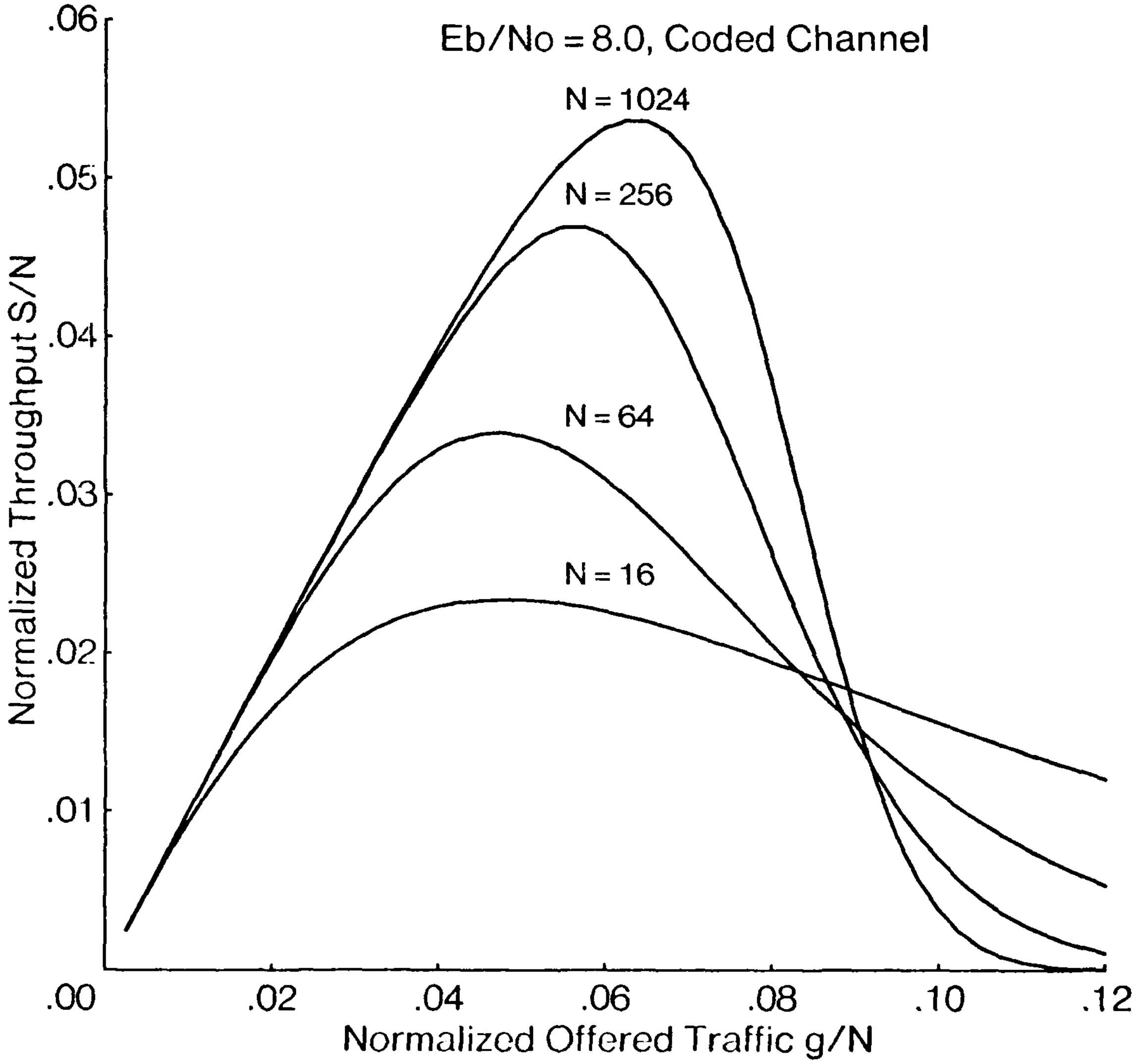

Figure 4.6. Normalized throughput S/N versus normalized offered traffic g/N for an infinite population.

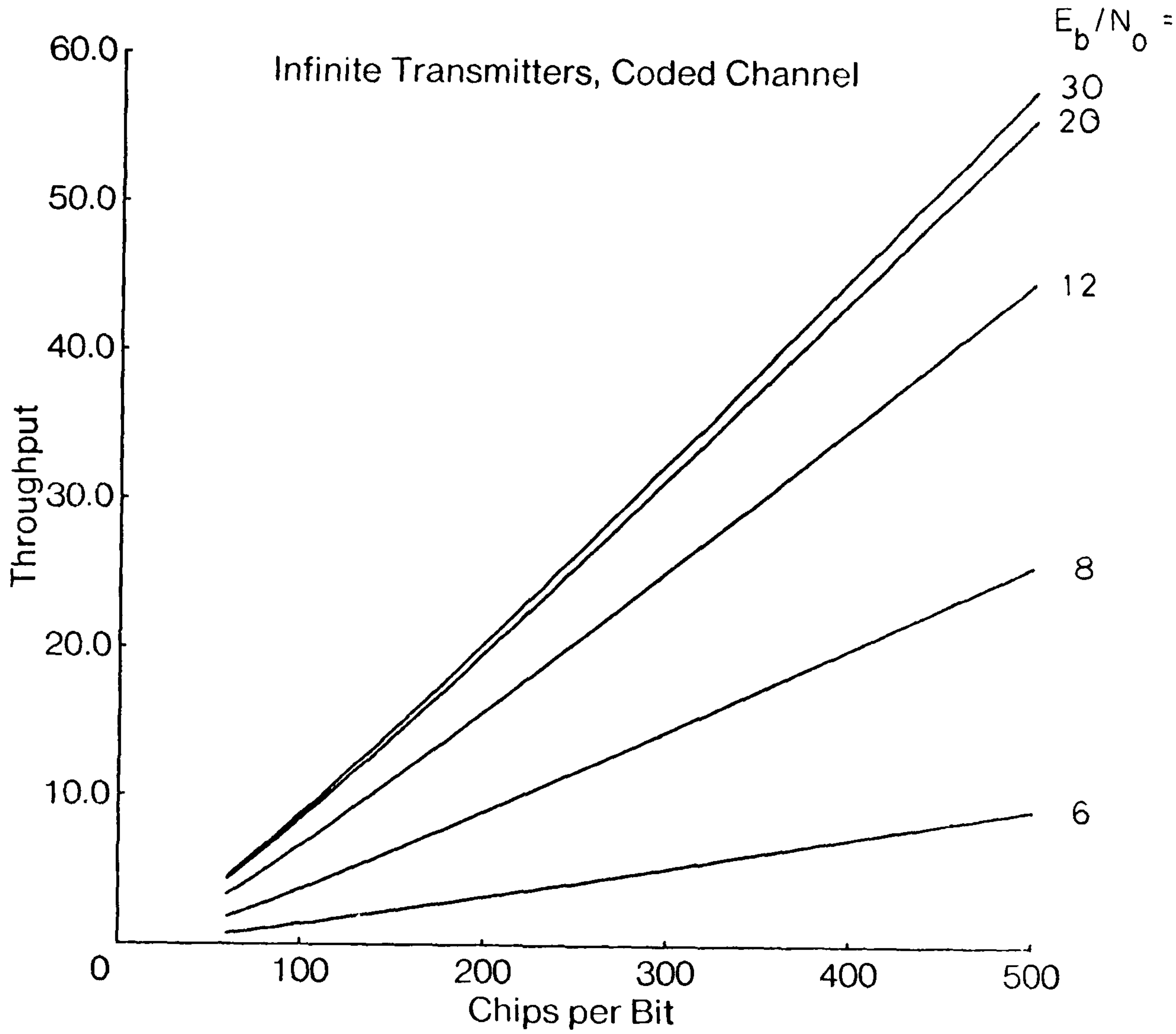

Figure 4.7. Throughput S versus chips per bit N for an infinite population.

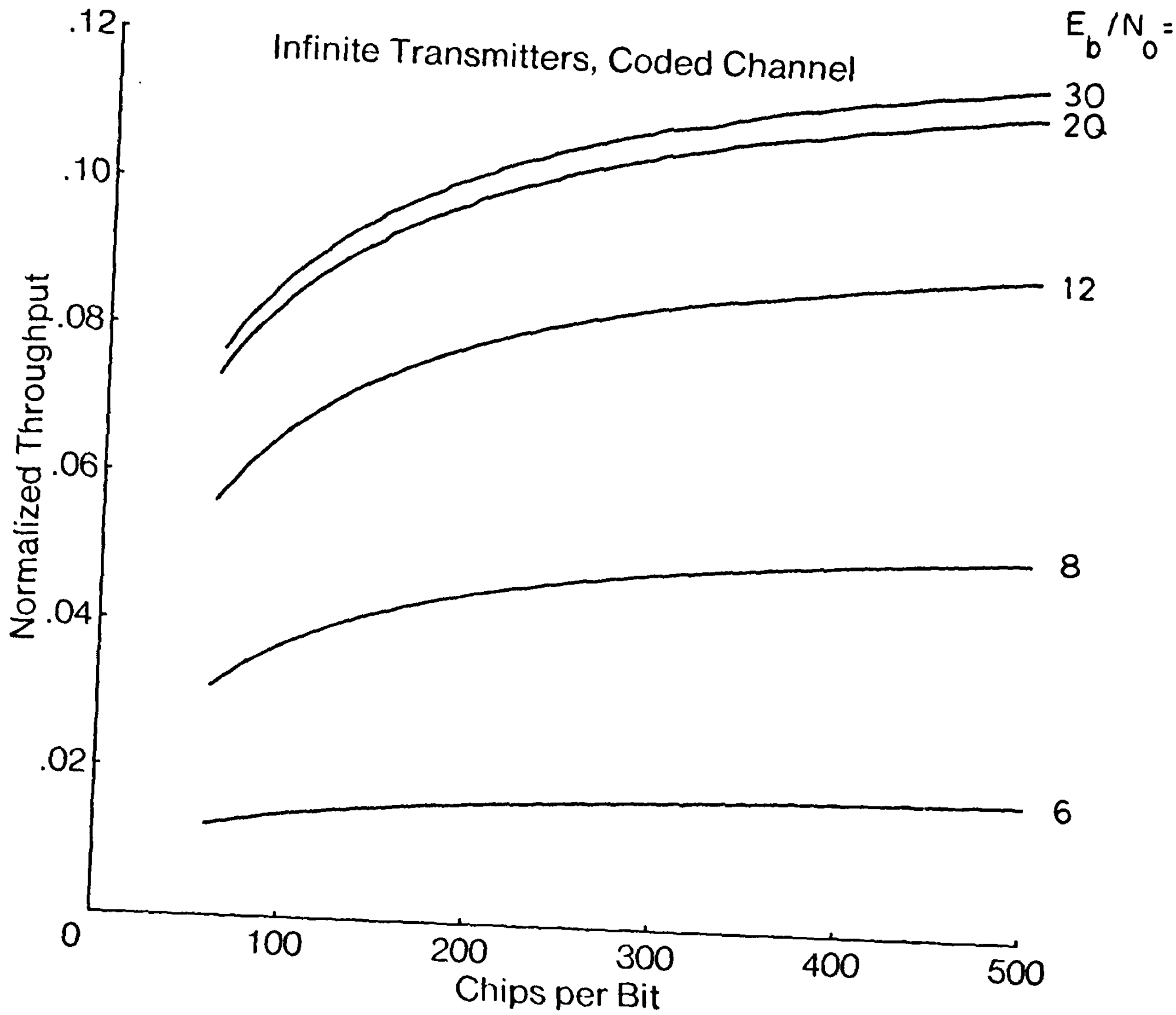

Figure 4.8. Normalized throughput S/N versus chips per bit N for an infinite population.

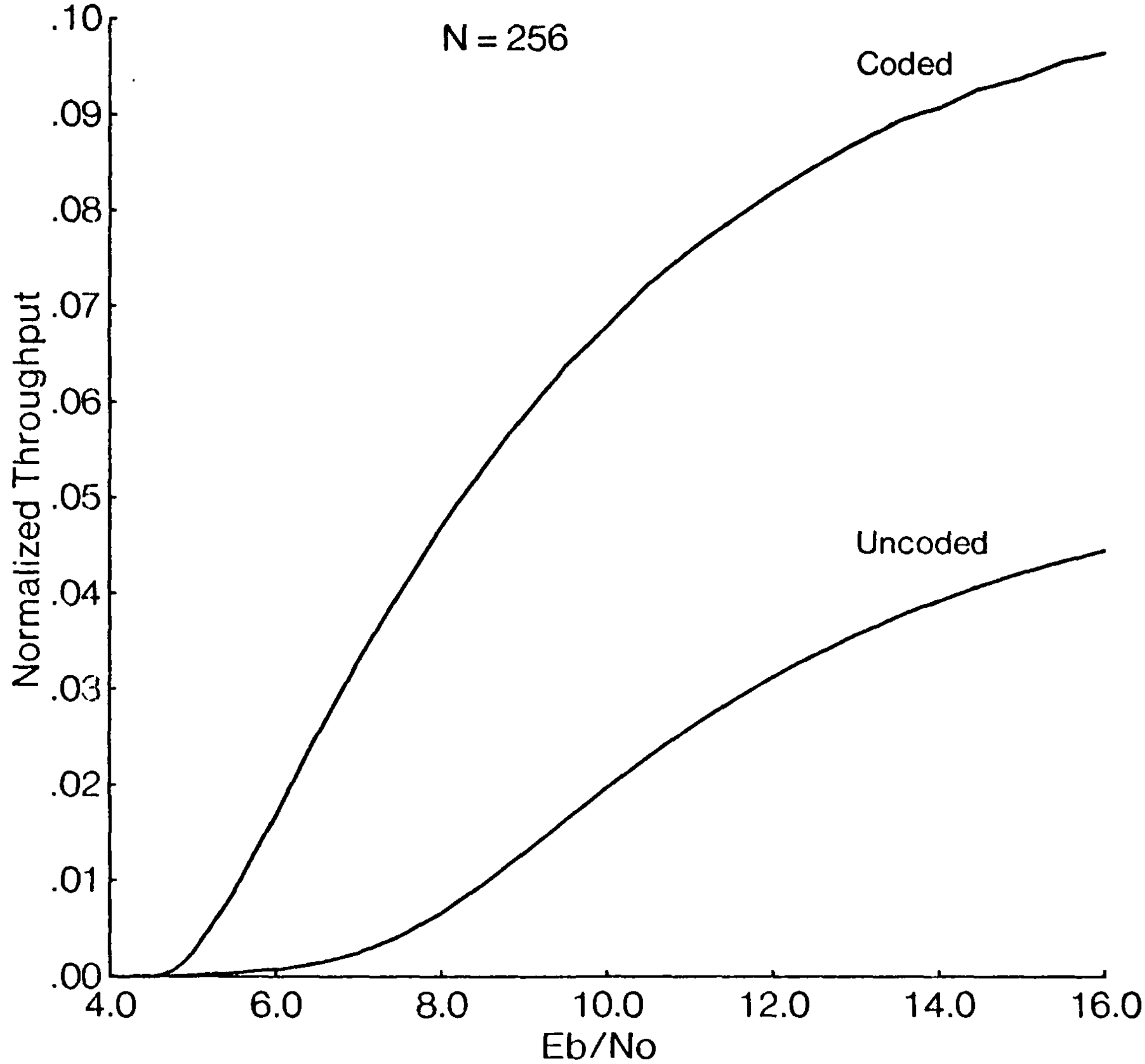

Figure 4.9. Normalized throughput S/N versus E_b/N_0 for an infinite population.

the curves flatten out for large E_b/N_0, as the multi-user interference begins to dominate the thermal noise.

Fig. 4.10 shows S vs. N for the infinite population case and for several values of M in the finite transmitters case, for $E_b/N_0 = 8.0$. As N increases, the channel reaches a point where $P_E(M)$ becomes small. At this point, even with all M users transmitting, there are few unsuccessful packets. Thus, increasing N further results in little increase in maximum throughput. This can be seen for the $M = 10$ and the $M = 20$ curves. Because of the additional bandwidth required, increasing N can actually cause a decrease in the normalized throughput. This is seen in Fig. 4.11. We note that the peak value of S/N for $M = 10$ and $M = 20$ is less than half the throughput for the uncoded $N = 1$ channel.

In Fig. 4.12, we plot the network throughput and also the individual user throughput vs. the number of users for the finite transmitters model, $E_b/N_0 = 8.0$ and $N = 128$ chips per bit. It is interesting to find that there is an optimum value of M which maximizes network throughput. We will see in Section 5 that this is no longer true when the effect of half-duplex radios is taken into account.

5. Half-Duplex Radios

5.A Analysis

We next consider a finite user network where both the sources and the destinations of packets are among the M users. Several modifications are needed. First, we must account for the case when the source is busy receiving a packet, and therefore will not transmit a packet which is scheduled. Second, we must account for the case when the destination is busy receiving or transmitting, and will not receive the packet. Finally, we must treat the effect on state transitions of an idle receiver not capturing a packet.

The state variable $\mathcal{X}(\tau)$ is now defined as both the number of active transmitters and the number of receivers which have captured a packet. This gives a state space $\mathcal{S} = \{(t, r) :$

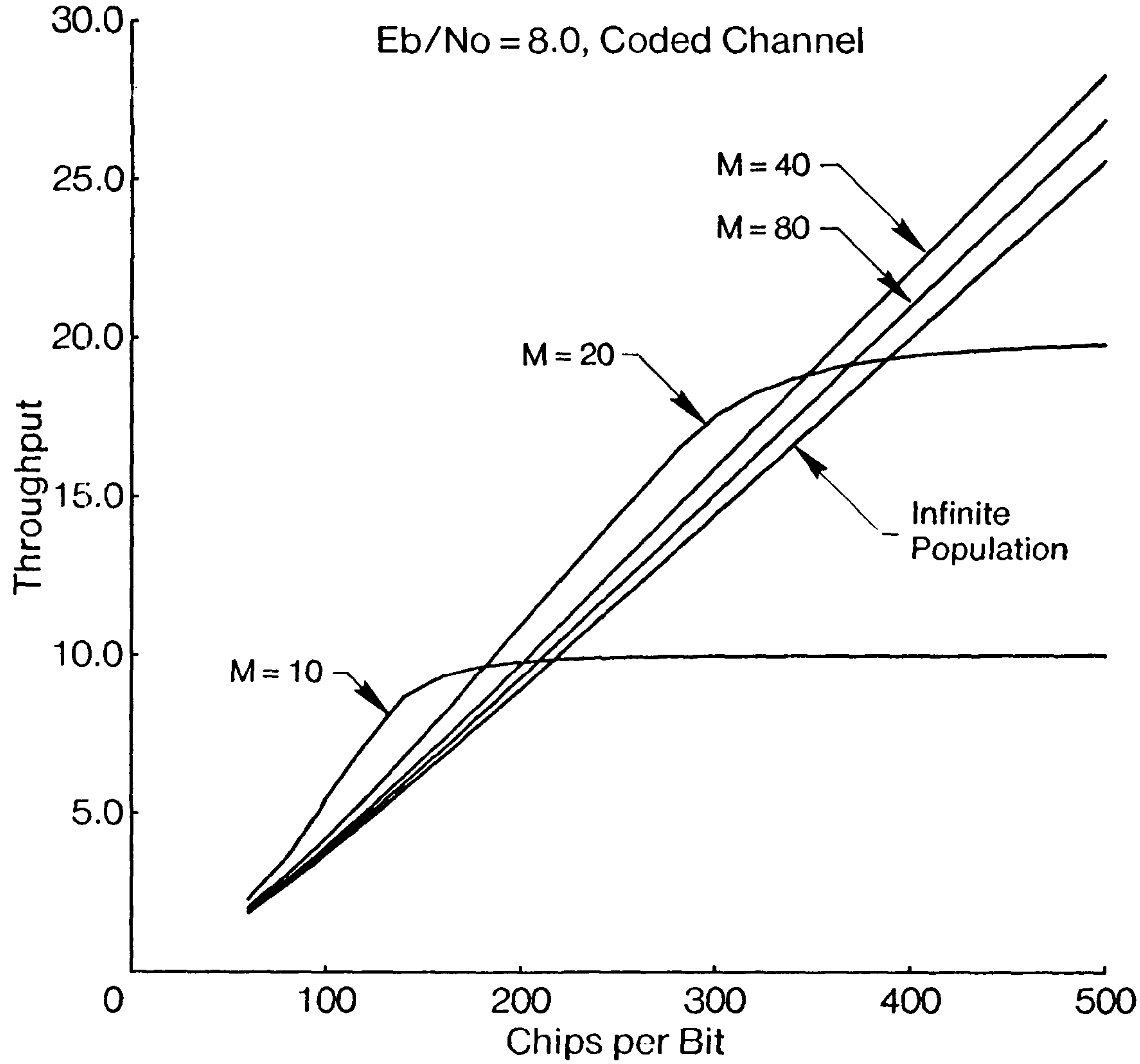

Figure 4.10. Throughput S versus chips per bit N for an infinite population and for finite transmitters.

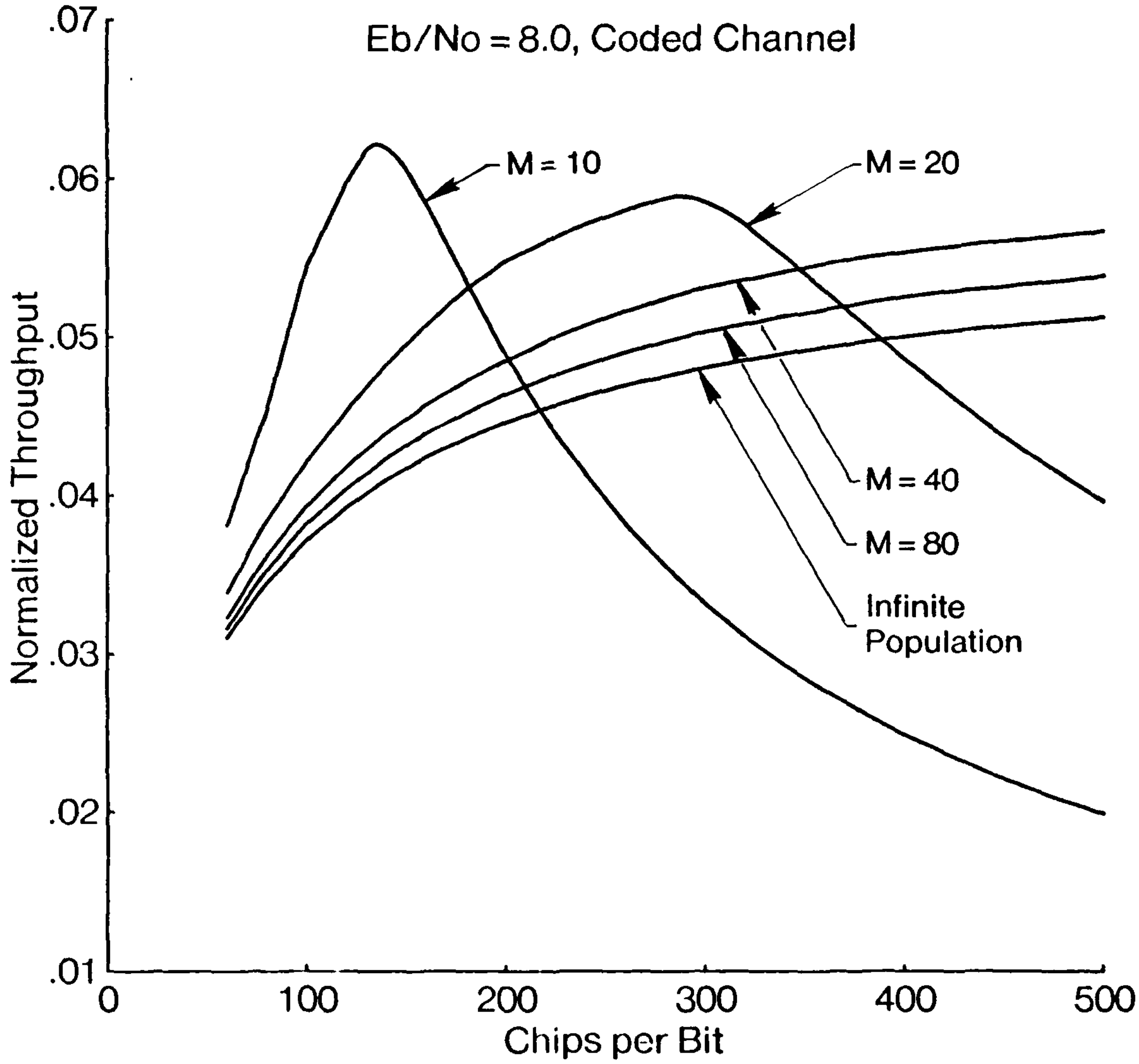

Figure 4.11. Normalized throughput S/N versus chips per bit N for an infinite population and for finite transmitters

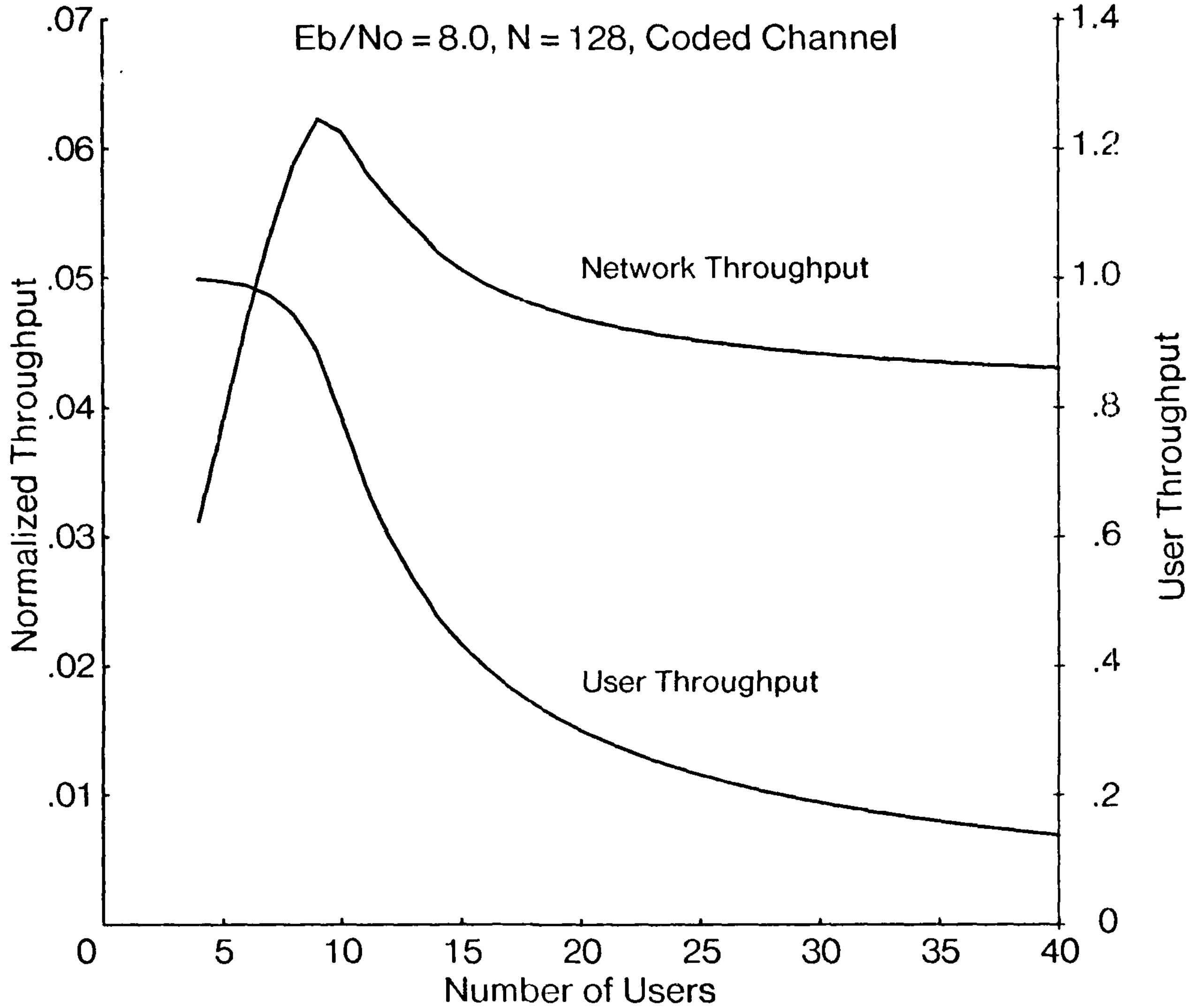

Figure 4.12. Throughput versus user population M for finite transmitters.

$0 \leq t \leq M, 0 \leq r \leq t,$ and $t + r \leq M\}$. For the basic model, we observed that successful transmissions could only begin when the state was in a subset S' of the state space S, corresponding to $j < L$. In the model with finite receivers, we find that this subset is now $S' = \{(t,r) : 0 \leq t \leq M-1, t < L, 0 \leq r \leq t,$ and $t + r \leq M - 1\}$. The non-absorbing states of the auxiliary Markov chain consist of the subset $\{(t,r) : (t-1, r-1) \in S'\}$.

We define the following variables:

$P_{S|(t,r)}$ is the probability that a transmitted packet is successful given that the network was in state (t,r) upon its arrival and given that it was captured.

$P_{SI}(t,r)$ is the probability that the source radio is idle given state (t,r)

$P_{DI}(t,r)$ is the probability that the destination radio is idle given state (t,r) and given that the source is idle

α_t is the probability that the destination radio synchronizes to the packet header given that it is idle and given that there are t interfering transmissions.

$\pi_{(t,r)}$ is the steady state distribution of $\mathcal{X}(\tau)$

$\tilde{\pi}_{(t,r)}$ is the steady state probability of the network being in state (t,r) at time τ_0^- given that a packet transmission begins at time τ_0

$E(T_s|(t,r))$ is the expected successful length as in the basic model.

$P_E(t)$ is the probability of first error for t simultaneous transmissions.

Because of identical users, for any user i, we find

$$P_{SI}(t,r) = \frac{M-t-r}{M} \tag{5.1}$$

$$P_{DI}(t,r) = 1 - \frac{t+r}{M-1} \tag{5.2}$$

since the destination is uniformly distributed over the $M-1$ other radios, $t+r$ of which are busy.

In state (t, r), transmissions are scheduled at rate $(M-t-r)\lambda$. A packet has a probability $\alpha_t P_{DI}(t, r)$ of being captured, which will cause a transition to state $(t+1, r+1)$. Otherwise, the packet will not be captured, so it will cause a transition to state $(t+1, r)$. Of the t packets being transmitted, r have captured a receiver and $t-r$ have not captured a receiver. Thus, due to packets transmissions completing, there is a transition to state $(t-1, r-1)$ at rate $r\mu$, and to state $(t-1, r)$ at rate $(t-r)\mu$. The state transitions for the general state (t, r) are diagrammed in Fig. 5.1. The forms of the Markov chain $\mathcal{X}(\tau)$ and of the auxiliary Markov chain are given in Figs. 5.2 and 5.3. The steady state probabilities $\pi_{(t,r)}$ of the Markov chain $\mathcal{X}(\tau)$ are not easily found analytically. However, they can be found from the conservation equations

$$\mathbf{Q}\underline{\pi} = 0 \tag{5.3}$$

where $\mathbf{Q}$ is the transition rate matrix, and from

$$\sum_{(t,r)\in S} \pi_{(t,r)} = 1 \tag{5.4}$$

Similar to the result in Section 3.A, we find

$$\begin{aligned} P_S &= \sum_{(t,r)\in S'} \tilde{\pi}_{(t,r)} \alpha_t P_{DI}(t, r) P_{S|(t,r)} \\ &= \frac{\sum_{(t,r)\in S'} P_{SI}(t, r)\pi_{(t,r)}\alpha_t P_{DI}(t, r) P_{S|(t,r)}}{\sum_{(t,r)\in S'} P_{SI}(t, r)\pi_{(t,r)}} \end{aligned} \tag{5.5}$$

As we show in Appendix E,

$$S_i = \sum_{(t,r)\in S'} \lambda \pi_{(t,r)} P_{SI}(t, r) P_{DI}(t, r) \alpha_t E(T_s|(t, r)) \tag{5.6}$$

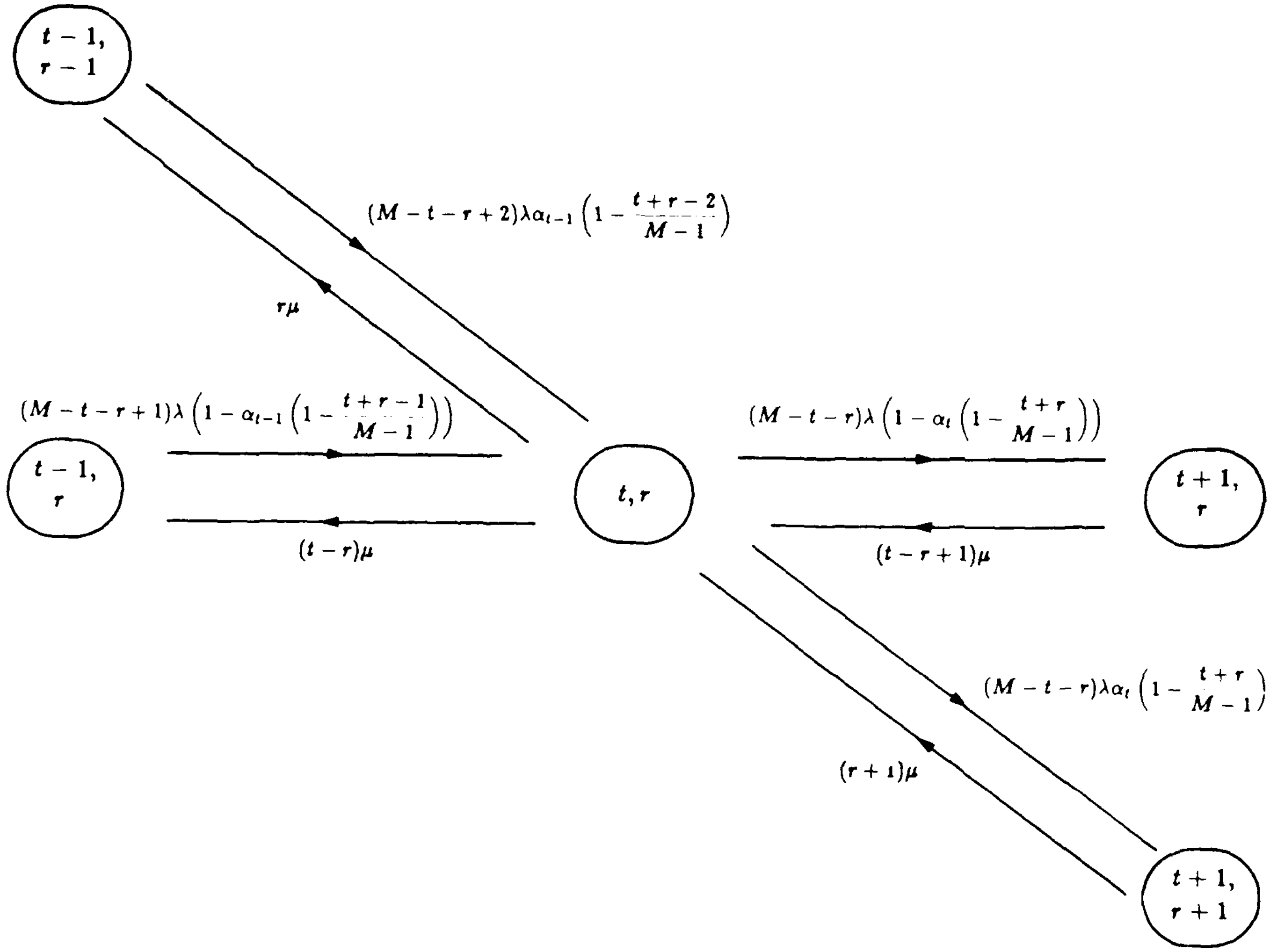

Figure 5.1. Local state-transition-rate diagram for the half-duplex model.

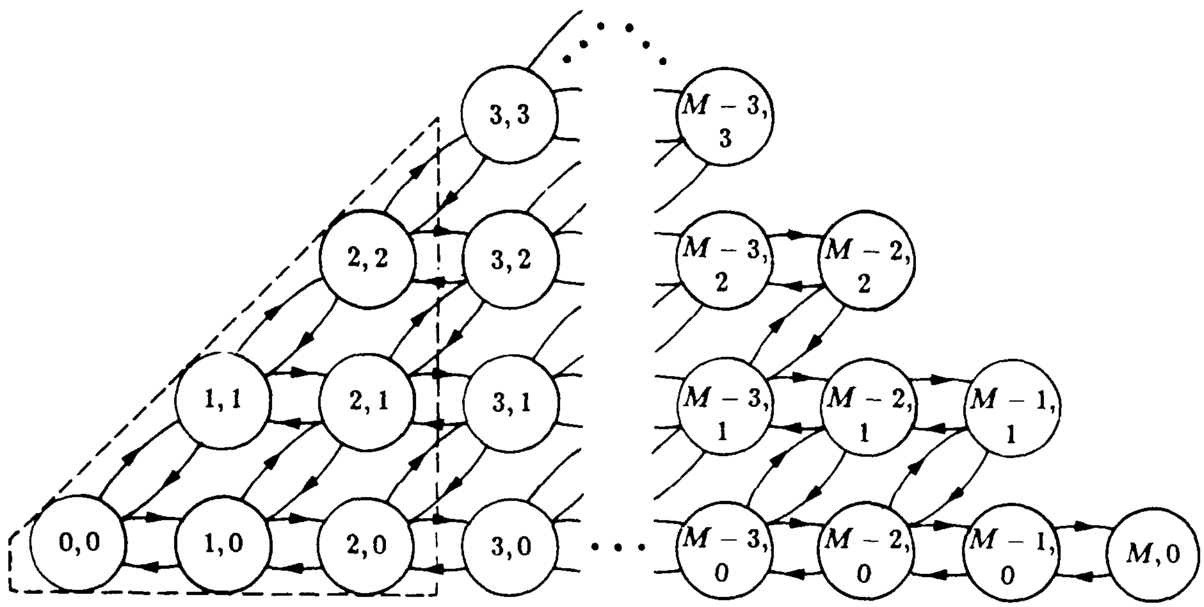

Figure 5.2. State-transition diagram for the half-duplex model.

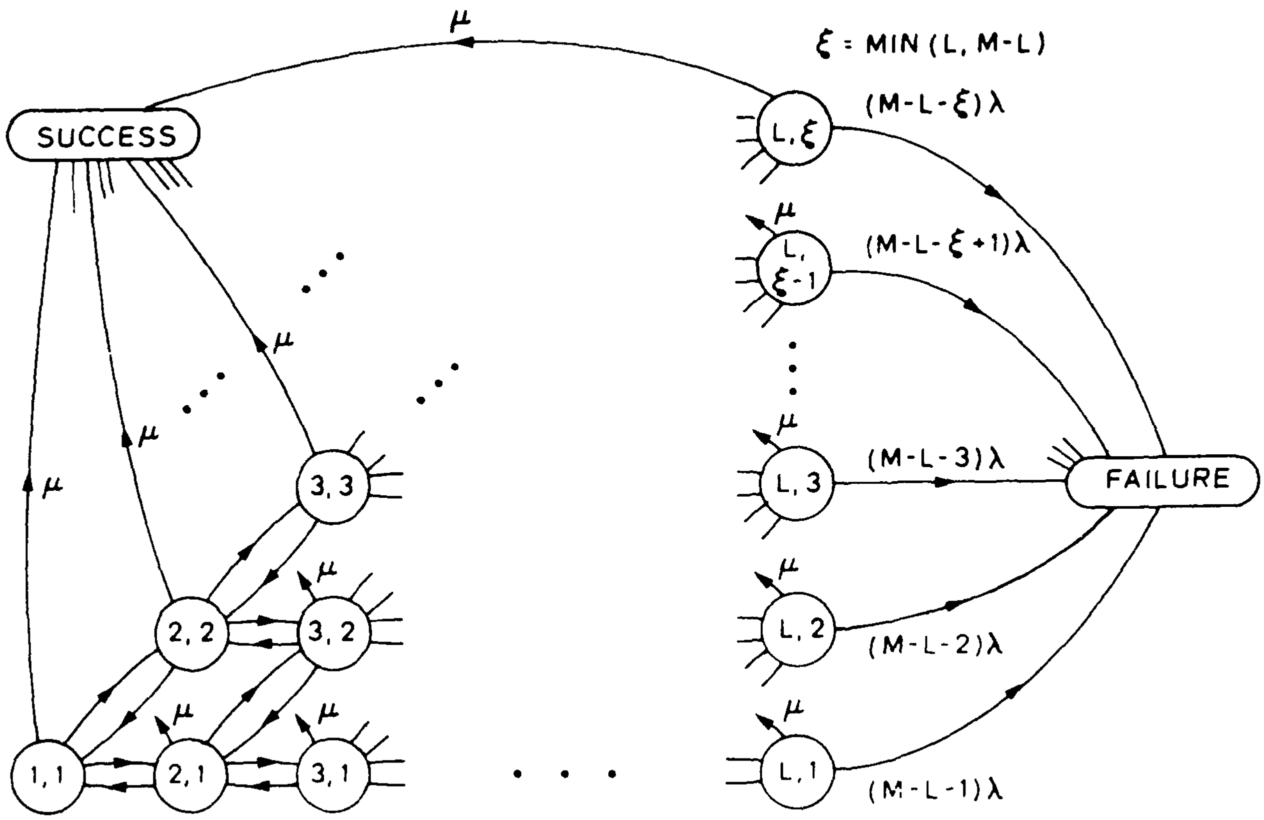

Figure 5.3. Auxiliary Markov chain for the half-duplex model.

Since $S = MS_i$, we find

$$S = \sum_{(t,r)\in S'} \lambda \pi_{(t,r)}(M - t - r)\left(1 - \frac{t+r}{M-1}\right)\alpha_t E(T_s|(t,r)) \tag{5.7}$$

$P_{S|(t,r)}$ and $E(T_s|(t,r))$ are again found from the auxiliary Markov chain transition rates. The Poisson error generation process described in Section 4 is included in this auxiliary Markov chain.

5.B Results

Throughput vs. N is plotted for several values of M in Fig. 5.4. We find that the throughput is much lower than for the same values of M in the basic model. This is expected, as the maximum possible throughput is $\lfloor M/2 \rfloor$. The jump at the point where the curve flattens out occurs when $L = M - 1$. This jump is due to the truncation of the decoder performance model for $P_E \geq 10^{-2}$. Below the jump, the decoder model uses the loose bound $P_E = 1.0$ for some states, while above the jump, the model uses a tighter bound for all states.

The maximum throughput flattens out around the point where the channel can support $M/2$ simultaneous transmissions with high probability of success. As t, the number of users transmitting, increases above $M/2$, the number of users that can be receiving is limited to $M - t$, so the probability that the destination is idle becomes very small. Thus, we expect that the optimum offered traffic is such that $t \leq \lfloor M/2 \rfloor$ for most of the time, so increasing N to support more than $\lfloor M/2 \rfloor$ transmissions should not have much impact on maximum throughput. In Fig. 5.5, we plot network throughput and also user throughput vs. M for $E_b/N_0 = 8.0$, $N = 128$. We find that, contrary to the earlier results of Section 4, the network throughput for the half-duplex model increases monotonically with M.

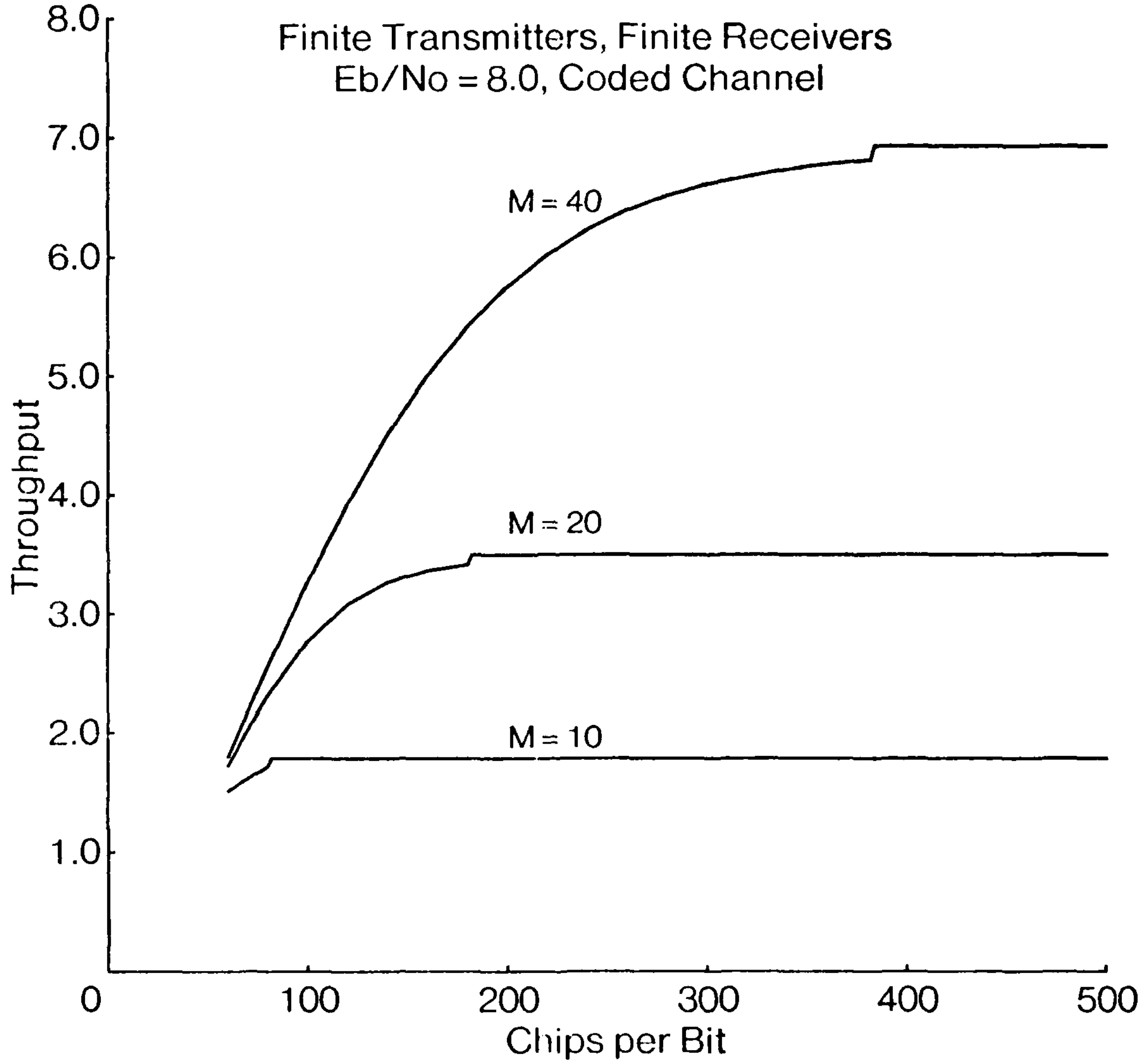

Figure 5.4. Throughput S versus chips per bit N

for the half-duplex model.

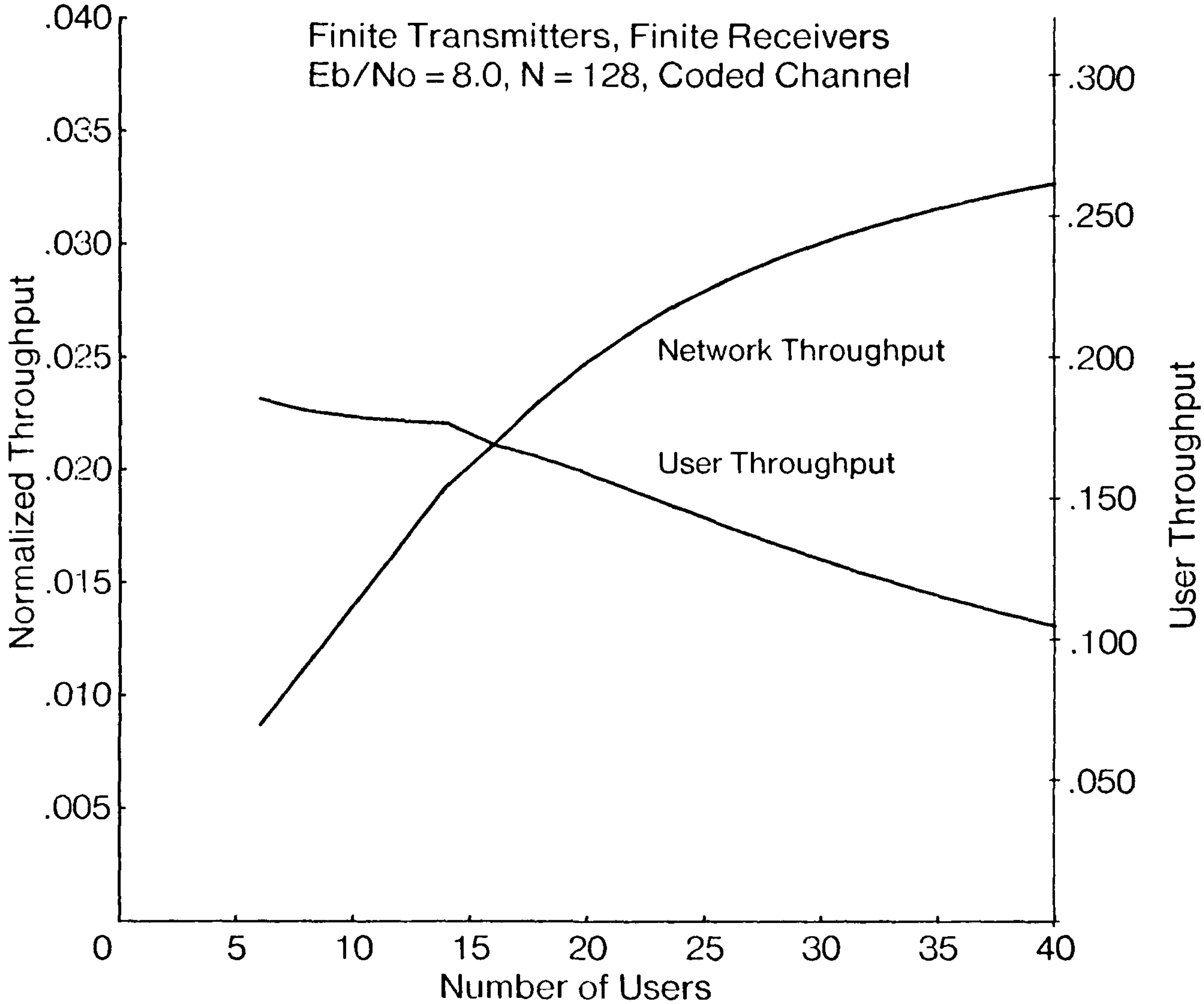

Figure 5.5. Throughput S versus user population M

for the half-duplex model.

6. Channel Load Sensing

Many authors have shown that for ALOHA-like systems, the network performance can be improved by sensing the channel and blocking transmissions when the channel is busy. In a CDMA network where multiple users can transmit simultaneously, similar improvement is possible if the radios can sense the number of radios transmitting and block transmission when the channel is heavily loaded. In addition to combatting multiple user interference, this technique of channel load sensing may also increase the robustness in the presence of jamming, by causing the channel capacity to degrade gracefully as the jamming signal power increases.

6.A Infinite Population

We use K to denote the number of sensed transmissions at which new transmissions are blocked. For $K > L$, the state space $\mathcal{S}$ is just a truncation of the basic model state space. This has the same state transition rates as the state space of the $M/M/K/K$ queue, which is a K server loss system [KLEI75]. The auxiliary Markov chain is the same as in the basic model; so for this range of $K > L$, the only difference from the basic model is in the initial state probabilities $\{\pi_j\}$, now given by

$$\pi_j = \frac{(\lambda/\mu)^j/j!}{\sum\limits_{k=0}^{K} (\lambda/\mu)^k/k!} \tag{6.1}$$

For very small K, almost all transmissions will be successful, because $P_E(j)$ will be small for all $j \in \mathcal{S}$. In the limit as $g \to \infty$, as soon as the end of a transmission is sensed a new packet transmission will begin. In the case of zero propagation delay, this will result in there always being exactly K users transmitting. The probability of success is the probability that the packet is completed before an error occurs. Thus, we find

$$P_{S|K-1,\tau} = e^{-\epsilon_K \tau} \tag{6.2}$$

$$E(T_S|K-1) = \frac{1}{(1+\epsilon_K/\mu)^2} \tag{6.3}$$

so

$$S_{\max} = \frac{K}{(1+\epsilon_K/\mu)^2} \tag{6.4}$$

6.B Half-Duplex Model

i) $K \geq M$

Because there will never be any more than M transmissions occurring, a channel load sense threshold $K \geq M$ will give the same performance as no channel load sensing.

ii) $K < M$ and ϵ_K/μ non-negligible

In this case, the state space is truncated at $t = K$, while the auxiliary Markov chain remains the same. Consequently, the only difference from the case studied in Section 5 is that the initial probability distribution $\{\pi_{(t,r)}\}$ changes. Thus, the solution can be found using the equations derived in Section 5.

iii) $K < M$ and $\epsilon_K/\mu << 1$

For this range of K, almost all captured packets will be successful. However, packets may not be captured, due to the destination radio being busy. Even so, we note that when K is small compared to M, the probability of no capture is very small, so the throughput is maximized by a very large g. Unfortunately, the numerical solution used for finding the throughput becomes unstable as g becomes large. Therefore, we use an approximation of the Markov chain which is valid for such values of g. For very large g, we expect the number of transmissions to be K, since as soon as a transmission ends a new scheduling point will be generated and transmission of a new packet will begin. However, in order to find the throughput, we must find the distribution of the number of captured receivers.

In this section, we first justify that the probability of there being fewer than K active transmitters is $\mathcal{O}(1/g)$. We then use an approximation of the Markov chain which contains only the states $\{(t,r) : t = K-1 \text{ or } t = K\}$ to find the distribution on the number of captured receivers. $E(T_S|(K-1,r)$ is found from Eqn. 3.21, to give

$$S = \sum_{r=0}^{K} \frac{r\pi_{(K,r)}}{(1+\epsilon_K/\mu)^2} \tag{6.5}$$

We denote the marginal probability of there being t users transmitting by $\hat{\pi}_t$, so

$$\hat{\pi}_t = \sum_{r=0}^{\xi} \pi_{(t,r)} \qquad \xi = \min(t, M-t) \tag{6.6}$$

We denote the expected number of users receiving given t users transmitting by $E(R|t)$, so

$$E(R|t) = \sum_{r=0}^{\xi} r \frac{\pi_{(t,r)}}{\hat{\pi}_t} \qquad \xi = \min(t, M-t) \tag{6.7}$$

We can find $\hat{\pi}_t$ from the balance equations resulting from a cut between states (t,r) and states $(t+1,r)$ (see Fig. 5.2).

$$\begin{aligned}\sum_{r=0}^{t}(M-t-r)\lambda\pi_{(t,r)} &= \sum_{r=0}^{t+1}(t+1)\mu\pi_{(t+1,r)} \\ (M-t)\lambda\hat{\pi}_t - \lambda\hat{\pi}_t E(R|t) &= (t+1)\mu\hat{\pi}_{t+1}\end{aligned} \tag{6.8}$$

so

$$\hat{\pi}_t = \frac{(M-(t-1)-E(R|t-1))\lambda}{t\mu}\hat{\pi}_{t-1} \tag{6.9}$$

If we define $E(I|t)$ as the expected number of idle radios given t users transmitting, we have

$$\begin{aligned}E(I|t) &= E(M-t-R|t) \\ &= M-t-E(R|t)\end{aligned} \tag{6.10}$$

so

$$\hat{\pi}_t = \frac{E(I|t-1)\lambda}{t\mu}\hat{\pi}_{t-1}$$

$$= \left(\prod_{u=0}^{t-1} \frac{(\lambda/\mu)E(I|u)}{(u+1)}\right)\hat{\pi}_0 \tag{6.11}$$

For $t < M/2$, there will always be at least one idle radio, so $1 \leq E(I|t) \leq M - t$. Therefore,

$$\frac{(\lambda/\mu)E(I|t)}{t+1} \qquad \text{is } \mathcal{O}(g) \text{ for } t < M/2, \tag{6.12}$$

which implies that for $K \leq M/2$, $\hat{\pi}_{K-1}$ is $\mathcal{O}\left(g^{-1}\right)$, and $\hat{\pi}_{K-2}$ is $\mathcal{O}\left(g^{-2}\right)$. Thus, to first order, for large g, we can consider the approximation to the Markov chain which is shown in Figs. 6.1 and 6.2.

The two cuts indicated in Fig. 6.2 give the following local balance equations.

$$(K-r)\mu\pi_{(K,r)} = \Lambda_r\left(1 - \alpha_{K-1}\left(1 - \frac{K+r-1}{M-1}\right)\right) \tag{6.13}$$

$$\Lambda_r = (K-r)\mu\pi_{(K,r)} + (r+1)\mu\pi_{(K,r+1)} \tag{6.14}$$

Where Λ_r is the rate of probability flowing out of state $(K-1, r)$, so

$$\Lambda_r = (M-K-r+1)\lambda\pi_{(K-1,r)} \tag{6.15}$$

From these equations, we find

$$(r+1)\mu\pi_{(K,r+1)} = \frac{(K-r)\mu\pi_{(K,r)}}{1-\alpha_{K-1}\left(1-\frac{K+r-1}{M-1}\right)} - (K-r)\mu\pi_{(K,r)} \tag{6.16}$$

so

$$\pi_{(K,r+1)} = \frac{K-r}{r+1}\left(\frac{1}{1-\alpha_{K-1}\left(1-\frac{K+r-1}{M-1}\right)} - 1\right)\pi_{(K,r)} \qquad 0 \leq r \leq K-1 \tag{6.17}$$

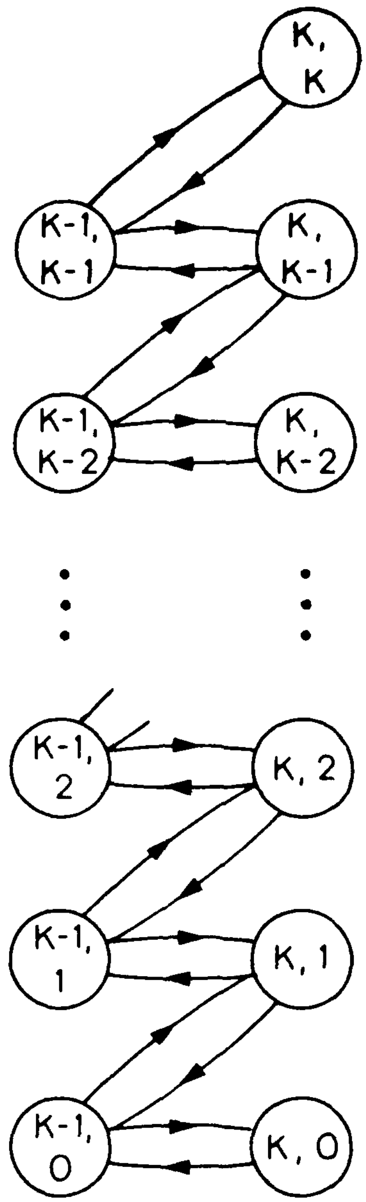

Figure 6.1. State-transition diagram for the half-duplex channel load sense model.

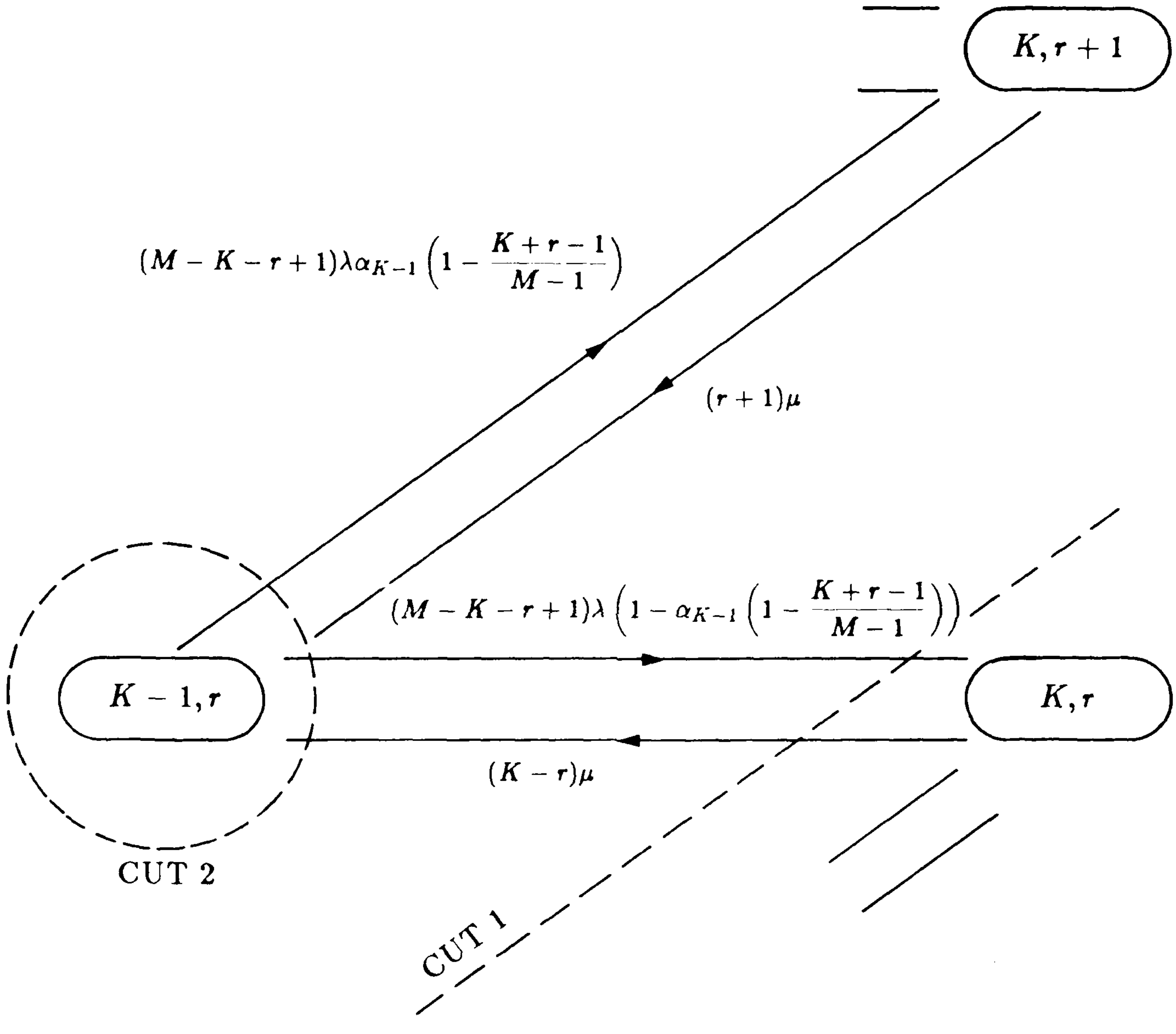

Figure 6.2. Local state-transition-rate diagram for the half-duplex channel load sense model.

From these recursion equations, we can find the probabilities $\pi_{(K,r)}$ in terms of $\pi_{(K,0)}$, and then normalize by $\sum \pi_{(K,r)} = 1$.

For the case of K small compared to M, and $\epsilon_K/\mu << 1$, we have found an approximation for the Markov chain which gives the distribution $\pi_{(K,r)}$, from which we find the throughput S. For larger K, the maximum throughput is achieved at values of g for which the approximation is no longer valid. However, for g in this range, we can use the numerical solution developed in Section 5.

6.C Propagation Delay

It is well known that carrier sensing systems degrade as the propagation delay becomes large relative to the packet lengths. To investigate this effect for the CDMA model, we introduce an exponentially distributed holding time at the beginning of each packet. The decision on whether or not to transmit is again based on the number of radios which can be sensed transmitting, but this decision does not account for users in the holding state. This model approximates the behavior due to propagation delay.

From the standpoint of a receiver, all other users have a state model which is diagrammed in Fig. 6.3(a). During the holding state, the receiver cannot detect the transmission. From the standpoint of a transmitter, the holding time and the transmission time are reversed. Thus, a packet transmission can be illustrated as in Fig. 6.3(b). The model has the peculiarity that a radio is busy both during the transmission time and during the holding time, so it is as if a radio continues to be busy from the time a packet transmission is completed until the time the packet finishes propagating to the receiver. This is shown as the shaded region in Fig. 6.3(b). We will see that this extra busy period causes the throughput to be reduced even for no channel load sensing.

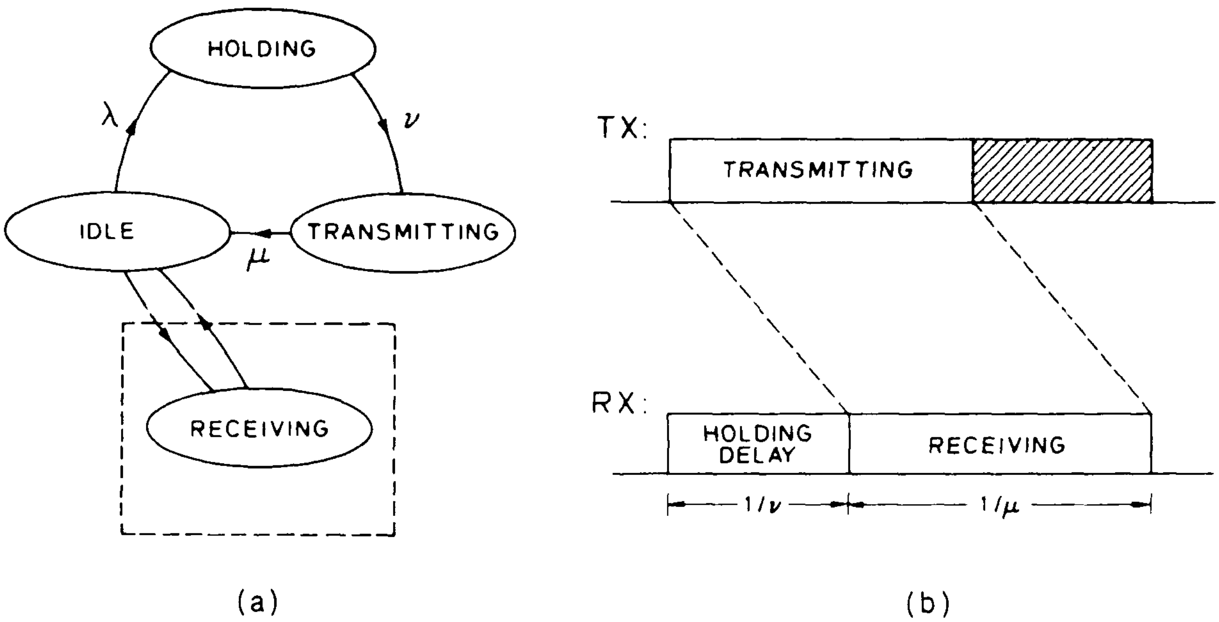

Figure 6.3. Exponential holding time model for non-zero propagation delay.

6.D Delay Model

We introduce the parameter ν, where $1/\nu$ is the average holding time. We also define h as ν/μ, so $1/h$ is the average holding time normalized by the mean packet length. This parameter $1/h$ is roughly equivalent to the standard delay measure a, which is the propagation delay normalized by the packet length. A radio can be either idle, holding, transmitting, or (for the half-duplex model) receiving, as we indicate in Fig 6.3(a).

We incorporate the holding time into the Markov chains as follows. Let w be the number of transmitters in a holding state. In the model which does not account for receivers, the states are indexed by w and t. The state space is $S = \{(w,t) : 0 \leq w \leq M, 0 \leq t \leq M - w\}$, and the subset from which successful transmissions can begin is $S' = \{(w,t) : 1 \leq w \leq M,\ 0 \leq t \leq M-w,\ \text{and } t < L\}$. For the state (w,t), $t < K$, transmissions are scheduled at rate $(M - w - t)\lambda$, causing transitions to state $(w+1,t)$. Users change from the holding state to the transmitting state at rate $w\nu$, causing transitions to state $(w-1,t+1)$. Transmissions are completed at rate $t\mu$, causing transitions to state $(w,t-1)$. The state transitions for the general state (w,t) are diagrammed in Fig. 6.4. For $t \geq K$, there are no transitions due to new packet transmissions, but the other transitions are not affected.

For the half-duplex model, we have a 3-dimensional state space indexed by w, t, and r;

$$S = \left\{ (w,t,r) : \begin{array}{l} 0 \leq w \leq M \\ 0 \leq t \leq M - w \\ 0 \leq r \leq M - w - t, \text{ and } r \leq t \,. \end{array} \right\}$$

$$S' = \left\{ (w,t,r) : \begin{array}{l} 1 \leq w \leq M \\ 0 \leq t \leq M - w, \text{ and } t < L \\ 0 \leq r \leq M - w - t, \text{ and } r \leq t \,. \end{array} \right\}$$

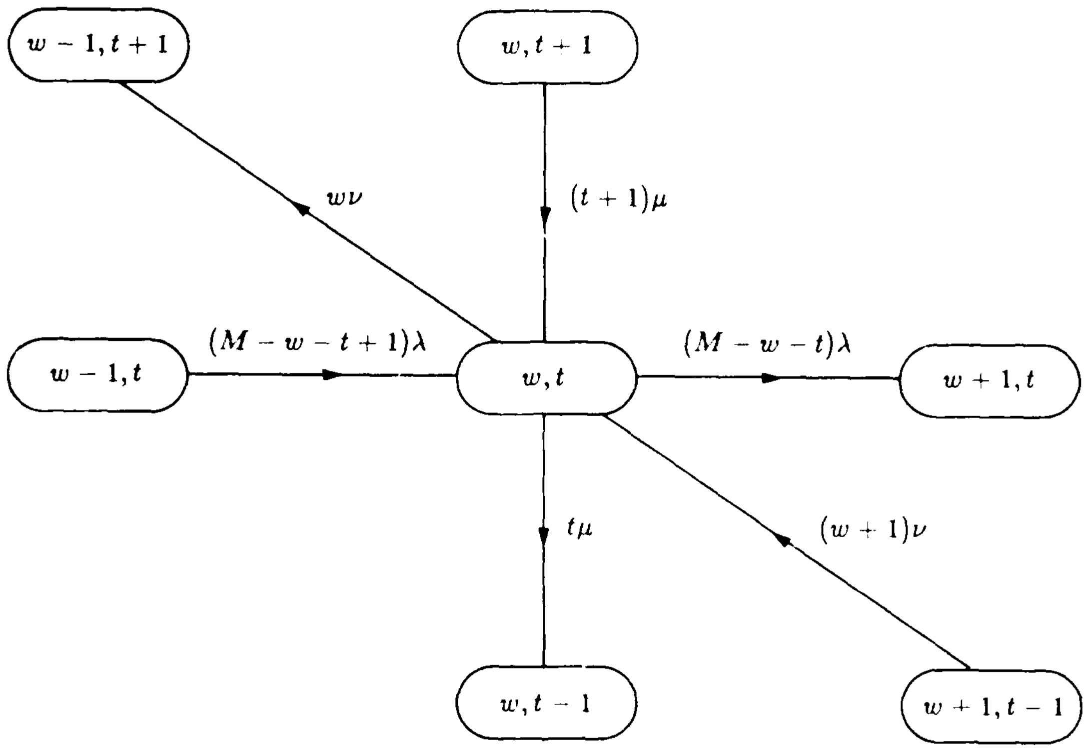

Figure 6.4. Local state-transition-rate diagram
for finite transmitters, non-zero propagation delay.

Once again, channel load sensing causes the deletion of transitions due to new packet transmissions for $t \geq K$.

The derivations of user throughput S_i are given in Appendix F. For the model which does not account for receivers, we find

$$S_i = \sum_{(w,t)\in S'} \nu\pi_{(w,t)} P_H(w,t)\alpha_t E(T_s|(w,t)) \tag{6.18}$$

where $P_H(w,t)$ is the probability that user i is in the holding state given state (w,t), which is simply w/M. So network throughput is

$$S = \sum_{(w,t)\in S'} \nu\pi_{(w,t)} w\alpha_t E(T_s|(w,t)) \tag{6.19}$$

Again, $E(T_s|(w,t))$ is found from $\mu\mathbf{R}^{-2}\underline{\mathbf{1}}$ for $t < L$. For the half-duplex model, we find

$$S_i = \sum_{(w,t,r)\in S'} \nu\pi_{(w,t,r)} P_H(w,t,r) P_{DI}(w,t,r)\alpha_t E(T_s|(w,t,r)) \tag{6.20}$$

where $P_H(w,t,r)$ is the probability that user i is in the holding state given state (w,t,r), which is simply w/M, and $P_{DI}(w,t,r)$ is the probability that the destination is idle given state (w,t,r) and given that i is holding, or

$$P_{DI}(w,t,r) = 1 - \frac{w+t+r-1}{M-1} \tag{6.21}$$

Thus,

$$S = \sum_{(w,t,r)\in S'} \nu\pi_{(w,t,r)} w \left(1 - \frac{w+t+r-1}{M-1}\right)\alpha_t E(T_s|(w,t,r)) \tag{6.22}$$

6.E Results

For an infinite population, $N = 256$, and $E_b/N_0 = 8.0$, we plot S/N vs. g for various values of K, and also for no channel load sensing in Fig. 6.5. For these parameters, the cut-off L is 26. It is seen that for small K, the normalized throughput increases towards K/N as g increases. Also, the maximum throughput for $K = 25$ is almost equal to the maximum for no channel load sensing. We then plot the throughput S vs. the channel load sense point K for an infinite population in Fig. 6.6, varying N from 64 to 1024. For N=64, channel load sensing can increase the throughput from 1.09 to 2.22, or 104%. The absolute increase is almost constant with respect to N, so for larger N, the percentage increase becomes small. Nevertheless, even though the increase in maximum throughput may be small, by using channel load sensing, the system can be made more stable. We note that in Fig. 6.5, the throughput varies only slightly over a wide range of offered traffic.

For the model of non-zero propagation delay, we plot the normalized throughput S/N vs. $1/h$ for the half-duplex model with $M = 12$. Results are shown in Fig. 6.7 for $N = 64$ and for $N = 128$ chips per bit, which give cut-offs of $L = 7$ and $L = 13$ respectively. These results were calculated for no channel load sensing and for K at the optimum channel load sense point. It is seen that even for no channel load sensing, the throughput decreases as $1/h$ increases, due to the users being busy during the holding periods. Also, as expected, the significant improvement from channel load sensing for $N = 64$ degrades as $1/h$ increases, and there is almost no improvement for $1/h = 1.0$.

7. Conclusions

In this report, we developed the Markovian model of a packet radio network for several specific fully connected topologies. We introduced a refinement which allows general bit error rate functions and incorporated a model of a DS-BPSK CDMA radio channel with convolutional FEC coding. Throughput and normalized throughput were found for various

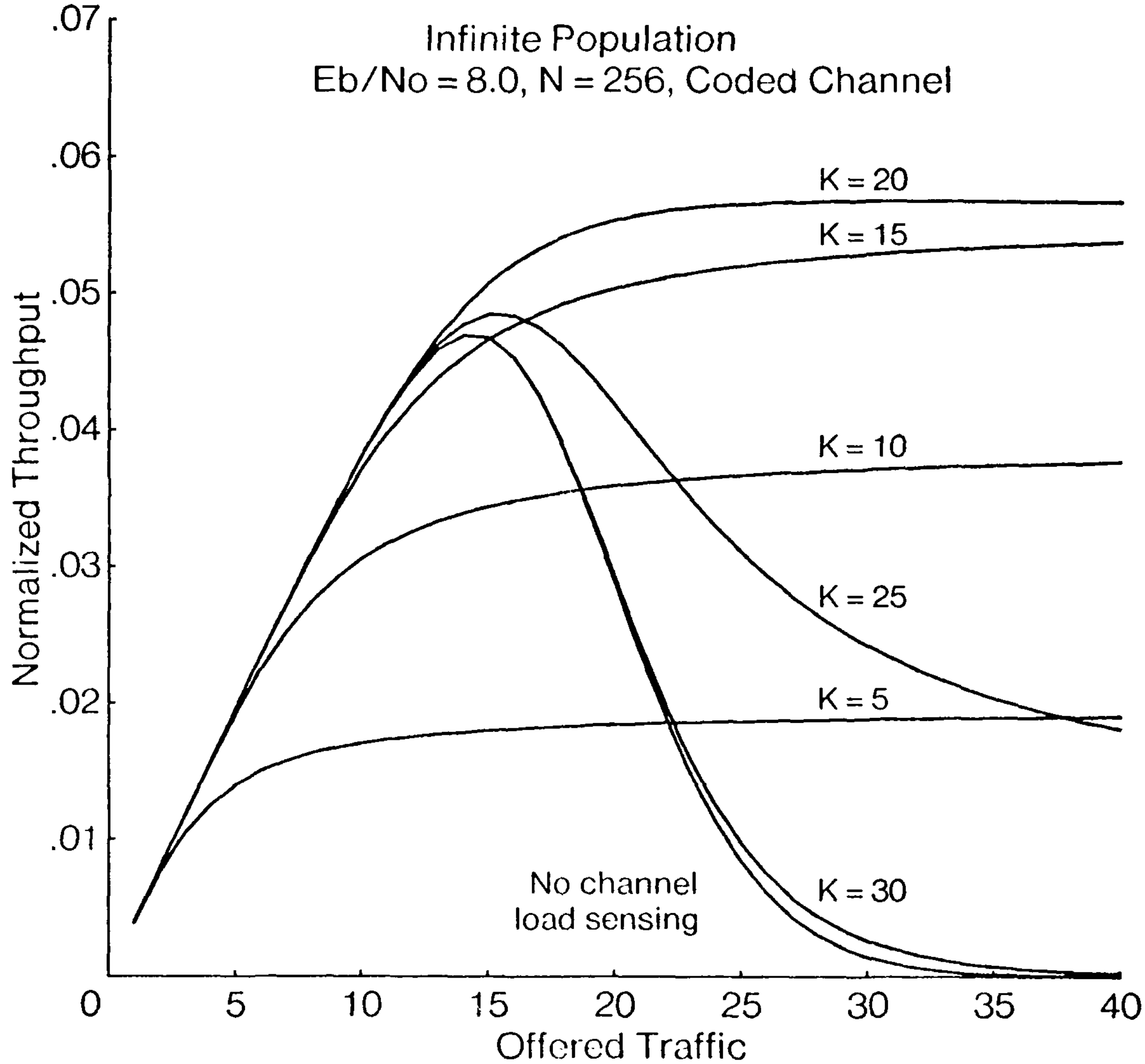

Figure 6.5. Throughput S versus offered traffic for several values of K for an infinite population with channel load sensing.

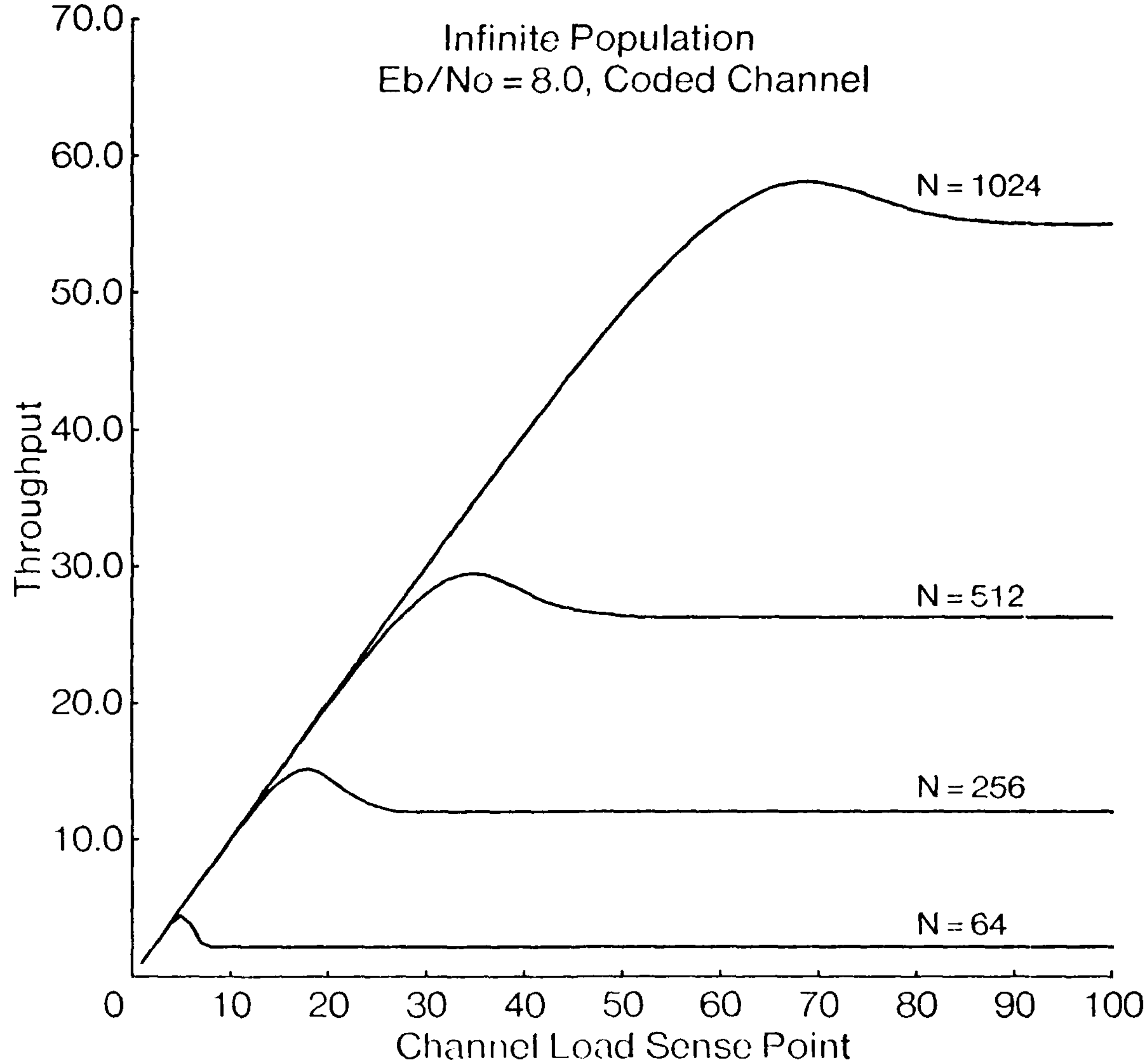

Figure 6.6. Throughput S vs. channel load sense point K for an infinite population.

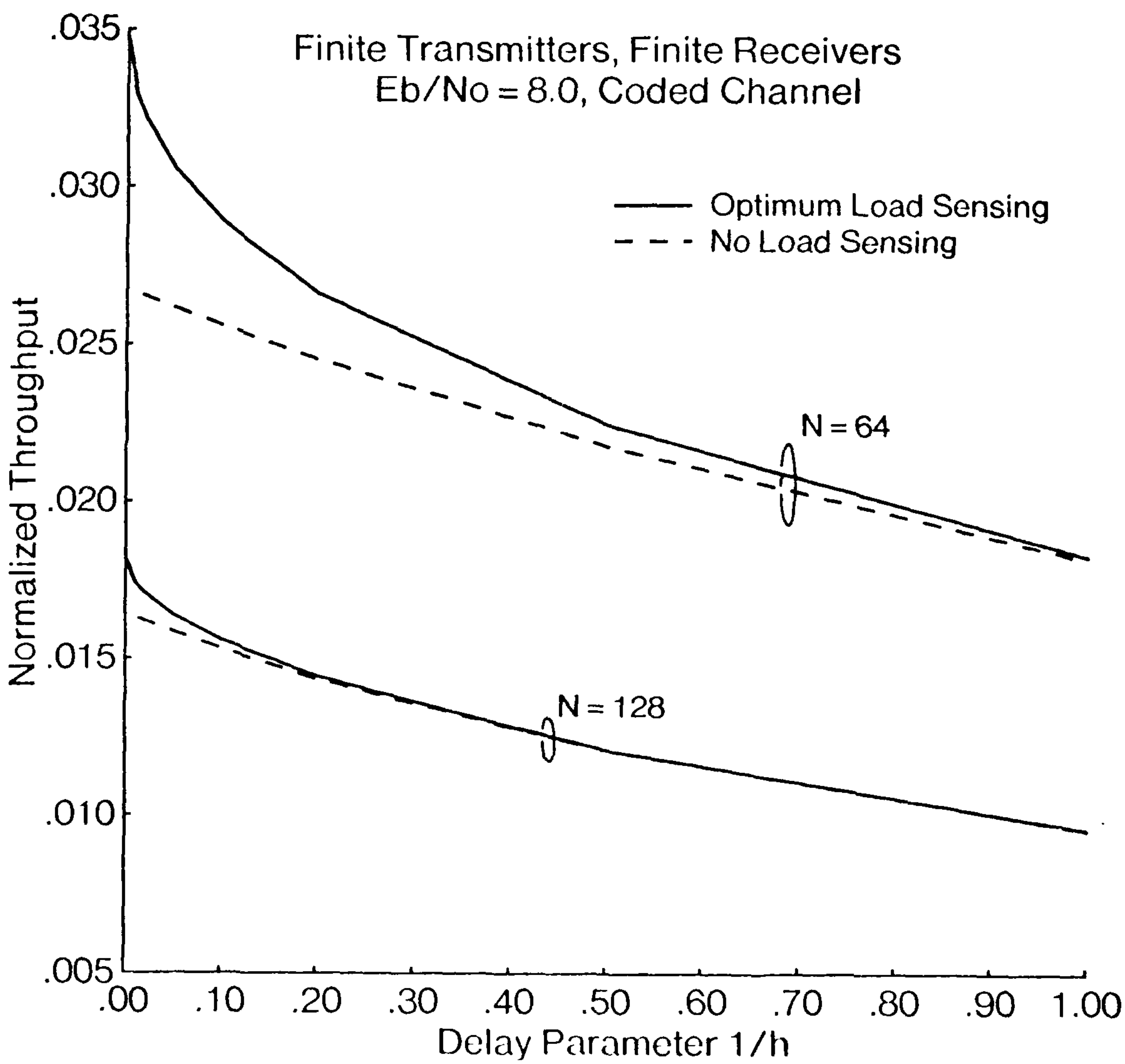

Figure 6.7. Normalized throughput S/N versus delay parameter $1/h$ for the half-duplex model with channel load sensing.

values of E_b/N_0 and chips per bit N. Next, we accounted for the effect of both the receivers and the transmitters belonging to the finite user population, as would be the case for a half-duplex radio network. This had a large impact on throughput. We then considered a channel load sensing protocol, and found that this resulted in increased throughput. Finally, a model of non-zero propagation delay was developed. We showed that increasing the delay degrades the improvement due to channel load sensing, a result familiar from CSMA analyses.

Appendix A. Throughput Equation For Finite Transmitters

To find the throughput for a user i, we expand the state space $\mathcal{X}(\tau)$ to explicitly indicate the state of user i, giving a new state space $\mathcal{X}^*(\tau)$

We define the following:

j^* is the number of users transmitting not including user i

c_i is the state of user i, with $c_i \in$ {idle, busy}, where i is busy when transmitting

S^* is the expanded state space, $S^* = \{(j^*, c_i) : 0 \leq j^* \leq M-1, \quad c_i \in \{\text{idle, busy}\}\}$

$\pi^*_{(j^*,c_i)}$ is the steady state probability of being in state (j^*, c_i)

$S(j^*, i)$ is the fraction of time that user i is successfully transmitting packets which found the network in state (j^*, idle) upon arrival.

The resulting Markov chain is shown in Fig. A1.

The analysis then follows exactly the solution to a more general case for multihop networks found by Brazio and Tobagi [BRAZ84]. First, we note that the times of successive transitions from the state (j^*, idle) to the state (j^*, busy) are regeneration points for the Markov chain $\mathcal{X}^*(\tau)$. Consider the cycles defined by the time intervals between two successive regeneration points. Let $C_k(j^*, i)$ denote the length of the kth cycle, and let $T_k(j^*, i)$ denote the successful length of the packet whose arrival initiated cycle k. The sequence

$$\{(C_k(j^*, i), T_k(j^*, i)) : k \geq 1\}$$

is a sequence of i.i.d. pairs of random variables. Let $E(C_k(j^*, i))$ and $E(T_k(j^*, i))$ denote the average values of $C_k(j^*, i)$ and $T_k(j^*, i)$ respectively. From the theory of renewal processes,

$$S(j^*, i) = \frac{E(T_k(j^*, i))}{E(C_k(j^*, i))} \qquad \text{with probability 1} \tag{A1}$$

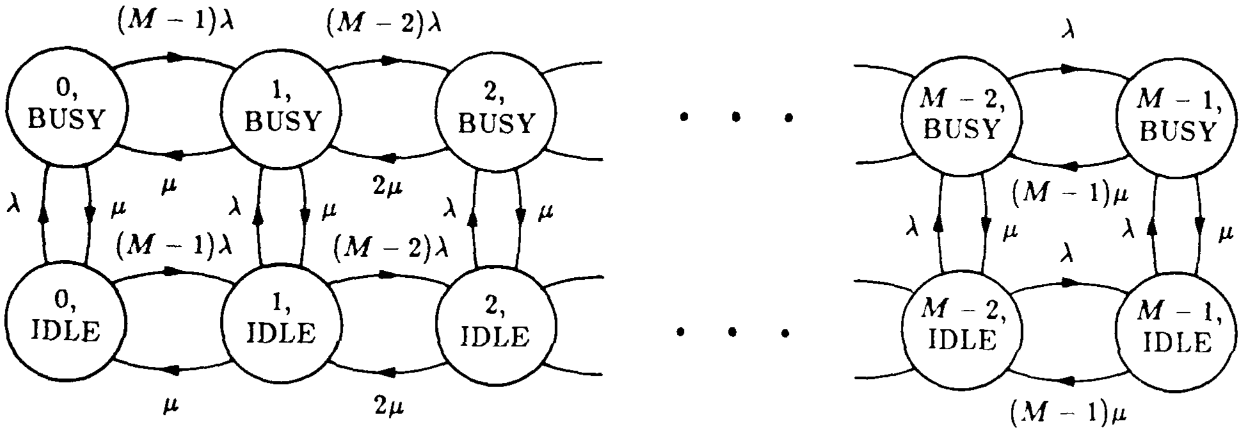

Figure A1. Expanded state-transition-rate diagram
for the finite transmitters model.

Also, as shown by Brazio and Tobagi in [BRAZ84],

$$E(C_k(j^*, i)) = \frac{1}{\lambda \pi^*_{(j^*,\ \text{idle})}} \tag{A2}$$

If user i is idle, and there are j^* other radios transmitting, the total number of users transmitting is j^*. Thus, the event $\{\mathcal{X}^* = (j^*, \text{idle})\}$ is the same as the event $\{\mathcal{X} = j \text{ and } i \text{ is idle})\}$, when $j^* = j$. In other words,

$$\pi^*_{(j^*,\ \text{idle})} = \pi_j P_{i_{IDLE}}(j) \qquad j^* = j \tag{A3}$$

so Eqn. A2 becomes

$$E(C_k(j^*, i)) = \frac{1}{\lambda \pi_j P_{i_{IDLE}}(j)} \qquad j^* = j \tag{A4}$$

Now, because all users are identical, the expected successful length of a packet that is transmitted given j^* other radios transmitting when it arrived does not depend on which particular radio i the packet arrived at. Consequently, $E(T_k(j^*, i))$ is simply $E(T_s|j)$, for $j^* = j$.

$$S(j^*, i) = \pi_j P_{i_{IDLE}}(j) E(T_s|j)) \qquad j^* = j \tag{A5}$$

The total throughput is simply the sum of conditional throughputs, so

$$S_i = \sum_{j^*=0}^{M-1} S(j^*, i) \tag{A6}$$

$$= \sum_{j=0}^{M-1} \pi_j P_{i_{IDLE}}(j) \lambda E(T_s|j) \tag{A7}$$

Finally, we notice that if $L < M$, $S(j^*, i) = 0$ for $j^* \geq L$, so less computation is required for smaller L.

Appendix B. Solution to $E(T_s|j)$

This derivation of $E(T_s|j)$ from the matrix $\mathbf{R}$ is also from [BRAZ84].

$\mathbf{P}^*(\tau)$ is the probability transition matrix of the auxiliary Markov chain, defined as

$$\mathbf{P}^*_{j,j'}(\tau) = \Pr(\mathcal{X}(\tau) = j' | \mathcal{X}(0) = j), \qquad j \in \mathcal{S}_{aux}, \qquad j' \in \mathcal{S}_{aux}$$

Similar to the corresponding rate transition matrix $\mathbf{R}^*$, $\mathbf{P}^*(\tau)$ has the form

$$\mathbf{P}^*(\tau) = \begin{pmatrix} \mathbf{P}(\tau) & \mathbf{P}_S(\tau) & \mathbf{P}_F(\tau) \\ \underline{\mathbf{0}} & 1 & 0 \\ \underline{\mathbf{0}} & 0 & 1 \end{pmatrix} \tag{B1}$$

Because of this structure, the forward Kolmogorov equations

$$\frac{d}{d\tau}\mathbf{P}^*(\tau) = \mathbf{P}^*(\tau) \cdot \mathbf{R}^* \qquad \mathbf{P}^*(0) = \mathbf{I} \tag{B2}$$

can be broken down to

$$\begin{aligned} \frac{d}{d\tau}\mathbf{P}(\tau) &= \mathbf{P}(\tau)\mathbf{R} & \mathbf{P}(0) &= \mathbf{I} \\ \frac{d}{d\tau}\mathbf{P}_S(\tau) &= \mu\mathbf{P}(\tau)\underline{\mathbf{1}} & \mathbf{P}_S(0) &= \underline{\mathbf{0}} \\ \frac{d}{d\tau}\mathbf{P}_F(\tau) &= \mathbf{P}(\tau)\varphi & \mathbf{P}_F(0) &= \underline{\mathbf{0}} \end{aligned} \tag{B3}$$

which has the solution

$$\begin{aligned} \mathbf{P}(\tau) &= e^{\mathbf{R}\tau} & \tau &\geq 0 \\ \mathbf{P}_S(\tau) &= \mu(e^{\mathbf{R}\tau} - \mathbf{I})\mathbf{R}^{-1}\underline{\mathbf{1}} & \tau &\geq 0 \\ \mathbf{P}_F(\tau) &= (e^{\mathbf{R}\tau} - \mathbf{I})\mathbf{R}^{-1}\varphi & \tau &\geq 0 \end{aligned} \tag{B4}$$

Since all the states except Success and Failure are transient, $e^{\mathbf{R}\tau} \to 0$ as $\tau \to \infty$. By construction, the $j + 1$st element of the column vector $\mathbf{P}_S(\infty)$ is the probability that a

transmitted packet which found the network in state j on arrival will be absorbed in the state Success, which we defined in Section 3.A as $P_{S|j}$. Thus, from Eqn. B4, we find $\mathbf{P}_S(\infty) = \mu(-\mathbf{I})\mathbf{R}^{-1}\underline{\mathbf{1}} = -\mu\mathbf{R}^{-1}\underline{\mathbf{1}}$. $\mathbf{P}_S(\infty)$ is equal to [illegible] from Section 3.A.

Now, let $\mathbf{T}$ be a random vector giving the time to absorption in Success. We construct $\mathbf{T}$ as column vector with rows indexed by the non-absorbing states in $\mathcal{S}_{aux}$ in the same order as the rows of $\mathbf{R}$. The $j+1$st element, $[\mathbf{T}]_{j+1}$, is the random variable which is the successful length of a packet which arrived to find the network in state j and was transmitted.

$$\sum_{j=0}^{L-1} \Pr([\mathbf{T}]_{j+1}) = \mathbf{P}_S(\tau) \qquad \tau \geq 0 \tag{B5}$$

and

$$\begin{aligned} E(\mathbf{T}|\mathbf{T} < \infty) &= \int_0^\infty (\mathbf{P}_S(\infty) - \mathbf{P}_S(\tau))\, d\tau \\ &= -\mu \int_0^\infty e^{\mathbf{R}\tau}\mathbf{R}^{-1}\,\underline{\mathbf{1}}\, d\tau = \mu\mathbf{R}^{-2}\underline{\mathbf{1}} \end{aligned} \tag{B6}$$

The desired quantity $E(T_s|j)$ is simply the $j+1$st element of this vector, so

$$E(T_s|j) = E\Big([\mathbf{T}]_{j+1} \;\Big|\; [\mathbf{T}]_{j+1} < \infty\Big) = [\mu\mathbf{R}^{-2}\underline{\mathbf{1}}]_{j+1} \tag{B7}$$

Appendix C. Viterbi Decoder performance

In this appendix, we summarize the analysis required to derive the performance of the decoder. The analysis is due mainly to Viterbi [VITE71]. Also, an excellent treatment of convolutional coding and of the Viterbi decoding algorithm is presented by Clark and Cain in [CLAR81].

C.1 Characteristics of Convolutional Codes

Unlike block codes, convolutional codes do not require a frame structure. Instead, the output symbols from the encoder at any time are a linear combination (modulo 2) of the previous k bits, so the output stream depends on the input bits in a sliding window fashion. For a rate m/n code, n symbols are output for each m bits. Typically, for $m = 1$ codes, a shift register will store the last k bits, and each of the n symbols will be generated by a different linear binary function of these bits.

The major parameters of the code are its constraint length k and its rate m/n. We only consider a commonly used rate 1/2 constraint length 7 code in the network model, but for illustrative purposes we will also discuss a rate 1/2 constraint length 3 code.

Convolutional codes are group codes, meaning that a set of symbols of a given length form a group. One convenient property of group codes is that for a binary symmetric channel (BSC), the probability of error does not depend on the actual data stream being sent. Thus, we can analyze the probability of error for the data stream of all zeros and this will yield the probability of error for any data stream. Furthermore, because the codes are linear, the all zero data stream will result in all output symbols being zero. For convenience, we denote the symbol stream of $m\mathcal{L}$ zeros by $C_0^{(\mathcal{L})}$. Also, without loss of generality, we assume that the initial loading of the encoder shift register is a sequence of $k - 1$ zeros.

C.2 Performance Analysis [VITE71]

Two performance measures are commonly derived for evaluating codes. The most widely used measure is P_B, the probability of bit error. This is the long term proportion of bit errors in a very long sequence of bits. Another measure is P_E, the probability of first error. This is the probability of a bit error at the output of the decoder, given that the first $k-1$ bits were known by the decoder, and given that no errors have occurred up to the present. In the network model, we assume that a known header precedes the packet transmission. This header will allow the decoder to be initialized to the correct state. Once initialized, we are interested in the first occurrence of an error in the decoder output bit stream, as this will cause the entire packet to be declared in error.

The Viterbi decoding algorithm is a maximum likelihood decision rule, in which the codeword which is closest in Hamming distance to the received codeword is chosen as the estimate of the transmitted symbol stream, and the corresponding bit stream is output as the estimate of the source's information stream. We denote the Hamming distance between $\mathcal{C}_\alpha$ and $\mathcal{C}_\beta$ by $d_H(\mathcal{C}_\alpha, \mathcal{C}_\beta)$. The probability of error given that $\mathcal{C}_0^{(\mathcal{L})}$ was sent is the probability that $\mathcal{R}^{(\mathcal{L})}$, the first $m\mathcal{L}$ received symbols, is closer to some other codeword of length $\mathcal{L}$ than it is to $\mathcal{C}_0^{(\mathcal{L})}$. This is the union of events,

$$\bigcup_{\alpha=1}^{2^{\mathcal{L}}-1} \left\{ d_H(\mathcal{R}^{(\mathcal{L})}, \mathcal{C}_\alpha^{(\mathcal{L})}) < d_H(\mathcal{R}^{(\mathcal{L})}, \mathcal{C}_0^{(\mathcal{L})}) \right\}$$

In the case of a tie between $\mathcal{C}_\alpha^{(\mathcal{L})}$ and $\mathcal{C}_0^{(\mathcal{L})}$, one is chosen at random. We can bound the probability of this event with the union bound, so that

$$\begin{aligned} \Pr(\text{error}) \leq & \sum_{\alpha=1}^{2^{\mathcal{L}}-1} \Pr\Big(d_H(\mathcal{R}^{(\mathcal{L})}, \mathcal{C}_\alpha^{(\mathcal{L})}) < d_H(\mathcal{R}^{(\mathcal{L})}, \mathcal{C}_0^{(\mathcal{L})}) \Big) \\ & + 1/2 \sum_{\alpha=1}^{2^{\mathcal{L}}-1} \Pr\Big(d_H(\mathcal{R}^{(\mathcal{L})}, \mathcal{C}_\alpha^{(\mathcal{L})}) = d_H(\mathcal{R}^{(\mathcal{L})}, \mathcal{C}_0^{(\mathcal{L})}) \Big) \qquad \text{(C1)} \end{aligned}$$

This union bound is the most common method used for analyzing decoder performance.

We can model the encoder as a finite state machine. Transitions between states occur at each new input bit. Because the oldest bit of the k bits in the shift register does not affect the next state, we only need 2^{k-1} states. The finite state machine representation of the rate $1/2$ $k=3$ encoder is shown in Fig. C1.

There is a one-to-one mapping between codewords and paths through the finite state machine. Consider two paths ξ_α and ξ_β of length $\mathcal{L}$, corresponding to codewords $\mathcal{C}_\alpha^{(\mathcal{L})}$ and $\mathcal{C}_\beta^{(\mathcal{L})}$, which both start in the state zero and end in the same state. Let $A^{(\mathcal{L})}$ be the set of all codewords of length $\mathcal{L}_1 > \mathcal{L}$ for which the first $\mathcal{L}$ bits are the codeword $\mathcal{C}_\alpha^{(\mathcal{L})}$. Every codeword $\mathcal{C}_\gamma^{(\mathcal{L}_1)} \in A^{(\mathcal{L}_1)}$ is the concatenation of $\mathcal{C}_\alpha^{(\mathcal{L})}$ and some sequence $\mathcal{C}_\psi^{(\mathcal{L}_2)}$, where $\mathcal{L} + \mathcal{L}_2 = \mathcal{L}_1$. Similarly, we form the set $B^{(\mathcal{L}_1)}$ from $\mathcal{C}_\beta^{(\mathcal{L})}$, and find that $\mathcal{C}_\delta^{(\mathcal{L}_1)} \in B^{(\mathcal{L}_1)}$ is the concatenation of $\mathcal{C}_\beta^{(\mathcal{L})}$ and $\mathcal{C}_\chi^{(\mathcal{L}_2)}$. Also, we denote the last $m\mathcal{L}_2$ symbols of $\mathcal{R}^{(\mathcal{L}_1)}$ by $\mathcal{R}_\omega^{(\mathcal{L}_2)}$.

We wish to find the sequences in $A^{(\mathcal{L})}$ and $B^{(\mathcal{L})}$ at minimum Hamming distance from $\mathcal{C}_0^{(\mathcal{L})}$. We note that

$$d_H(\mathcal{R}^{(\mathcal{L}_1)}, \mathcal{C}_\gamma^{(\mathcal{L}_1)}) = d_H(\mathcal{R}^{(\mathcal{L})}, \mathcal{C}_\alpha^{(\mathcal{L})}) + d_H(\mathcal{R}_\omega^{(\mathcal{L}_2)}, \mathcal{C}_\psi^{(\mathcal{L}_2)}) \tag{C2}$$

and

$$d_H(\mathcal{R}^{(\mathcal{L}_1)}, \mathcal{C}_\delta^{(\mathcal{L}_1)}) = d_H(\mathcal{R}^{(\mathcal{L})}, \mathcal{C}_\beta^{(\mathcal{L})}) + d_H(\mathcal{R}_\omega^{(\mathcal{L}_2)}, \mathcal{C}_\chi^{(\mathcal{L}_2)}) \tag{C3}$$

Let $\hat{\mathcal{C}}_\gamma^{(\mathcal{L}_1)} \in A^{(\mathcal{L}_1)}$ be the codeword which minimizes $d_H(\mathcal{R}^{(\mathcal{L}_1)}, \mathcal{C}_\gamma^{(\mathcal{L}_1)})$, and similarly for $\hat{\mathcal{C}}_\delta^{(\mathcal{L}_1)} \in B^{(\mathcal{L}_1)}$. Because the starting state is the same, $\mathcal{C}_\psi^{(\mathcal{L}_2)}$ and $\mathcal{C}_\chi^{(\mathcal{L}_2)}$ range over the same sequences. This implies that $d_H(\mathcal{R}_\omega^{(\mathcal{L}_2)}, \hat{\mathcal{C}}_\psi^{(\mathcal{L}_2)})$ must equal $d_H(\mathcal{R}_\omega^{(\mathcal{L}_2)}, \hat{\mathcal{C}}_\chi^{(\mathcal{L}_2)})$. Therefore, the codeword which is uniquely closest to $\mathcal{R}^{(\mathcal{L}_1)}$ will be in $A^{(\mathcal{L}_1)}$ only if $d_H(\mathcal{R}^{(\mathcal{L})}, \mathcal{C}_\alpha^{(\mathcal{L})}) < d_H(\mathcal{R}^{(\mathcal{L})}, \mathcal{C}_\beta^{(\mathcal{L})})$, and will be in $B^{(\mathcal{L}_1)}$ only if $d_H(\mathcal{R}^{(\mathcal{L})}, \mathcal{C}_\alpha^{(\mathcal{L})}) > d_H(\mathcal{R}^{(\mathcal{L})}, \mathcal{C}_\beta^{(\mathcal{L})})$. In the case of a tie, $d_H(\mathcal{R}^{(\mathcal{L})}, \mathcal{C}_\alpha^{(\mathcal{L})}) = d_H(\mathcal{R}^{(\mathcal{L})}, \mathcal{C}_\beta^{(\mathcal{L})})$, the codewords $\hat{\mathcal{C}}_\gamma^{(\mathcal{L}_1)}$ and $\hat{\mathcal{C}}_\delta^{(\mathcal{L}_1)}$ will have equal

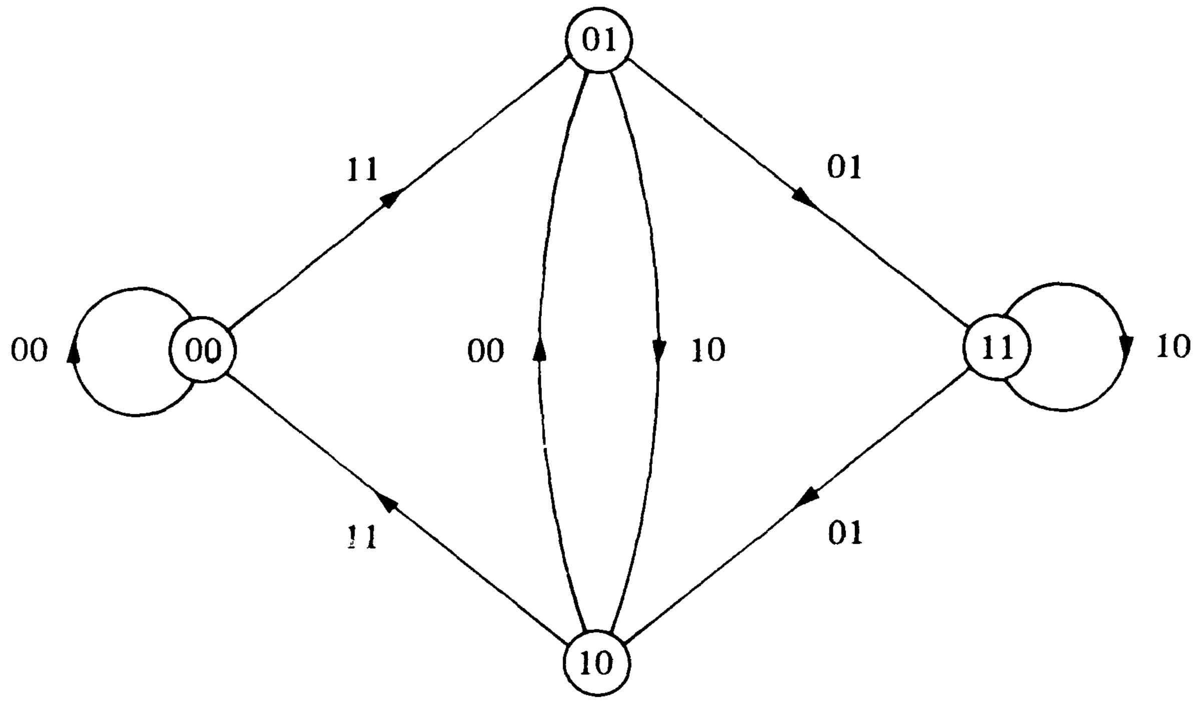

Figure C1. Finite state machine model.

Rate 1/2 k=3 code.

Hamming distances to $\mathcal{C}_0^{(\mathcal{L}_1)}$, so one must be chosen at random. We can make this random choice at bit $\mathcal{L}$ without increasing the probability of error. Therefore, no matter how the paths compare, at bit $\mathcal{L}$ we can eliminate one of the paths entering a given state. In fact, we can eliminate all but one path entering every state. We call the remaining paths the survivors.

Because of the known starting state, during the first $k - 2$ bits, there are a limited number of states that the encoder can be in. For example, it cannot be in the all ones state since all of the zeros will not have been shifted out. After $k - 1$ bits, it can be in any state, so the k^{th} bit can be mapped into any of the 2^k transitions. There are at most 2^{k-1} survivors. At each bit a comparison must be made between the two paths entering every state, or at most 2^k comparisons. The decoder must retain the 2^{k-1} survivors and the Hamming distance between the received symbol sequence and each of the 2^{k-1} survivors.

For a finite length message, a known sequence of $k - 1$ bits must be appended. This will give a uniquely specified ending state, so the decoder will be able to decide between the 2^{k-1} survivors.

Assuming that the all zero codeword $\mathcal{C}_0^{(\mathcal{L})}$ was sent, an error event occurs at bit $\mathcal{L}$ when the path merging with the state zero is chosen as a survivor, eliminating $\mathcal{C}_0^{(\mathcal{L})}$. For a fixed length message, if we also assume that the known concluding bits are a sequence of $k - 1$ zeros, the decoder will always choose a codeword which ends in the state zero, so again the only comparisons we need to consider are those between $\mathcal{C}_0^{(\mathcal{L})}$ and the path entering the state zero. Denote the codeword corresponding to the path entering the state zero at bit $\mathcal{L}$ by $\mathcal{C}_E^{(\mathcal{L})}$. The difference in Hamming distance between $\mathcal{R}^{(\mathcal{L})}$ and $\mathcal{C}_0^{(\mathcal{L})}$ and $\mathcal{R}^{(\mathcal{L})}$ and $\mathcal{C}_E^{(\mathcal{L})}$ only depends on those symbols of $\mathcal{R}^{(\mathcal{L})}$ for which $\mathcal{C}_E^{(\mathcal{L})} = 1$. Let i be the number of symbols for which $\mathcal{C}_E^{(\mathcal{L})} = 1$. The probability that $\mathcal{C}_E^{(\mathcal{L})}$ is chosen as the survivor is P_i, the

probability that more than half of these i symbols are in error, which is

$$P_i = \begin{cases} \displaystyle\sum_{e=(i+1)/2}^{i} \binom{i}{e} p^e (1-p)^{i-e}, & i \text{ odd;} \\ \displaystyle\frac{1}{2}\binom{i}{i/2} p^{i/2}(1-p)^{i/2} + \sum_{e=i/2+1}^{i} \binom{i}{e} p^e (1-p)^{i-e}, & i \text{ even.} \end{cases} \quad \text{(C4)}$$

where $p = \overline{P}_{e,symbol}$, the channel symbol error rate.

We wish to find the number of codewords which begin in state zero, return to state zero for the first time at bit $\mathcal{L}$ and have i ones. This is done by modifying the state diagram as follows. Split the zero node into a source node and a sink node. Assign a gain of $LN^{\beta}D^{\gamma}$ to the transitions, where β is the weight of the input bit (i.e., 0 for a 0, 1 for a 1), and γ is the weight of the output bit (i.e., 0 for 00, 1 for 01 or 10, and 2 for 11). For a path of length α' with β' ones in the input bits and γ' ones in the output symbols, the product of the transition gains is $L^{\alpha'}N^{\beta'}D^{\gamma'}$. The modified state diagram is given in Fig. C2.

If we sum together all possible paths from the source to the sink, we will find the overall gain of the network, $T(D, L, N)$. For example, for the rate 1/2 $k = 3$ code, this is

$$T(D,L,N) = D^5L^3N + D^6L^4(1+L)N^2 + \ldots + D^{5+j}L^{3+j}(1+L)^jN^{1+j} + \ldots \quad \text{(C5)}$$

This gain can be found directly from the modified state diagram by solving the set of linear equations giving the dependencies between the nodes. For example, for the rate 1/2 $k = 3$ code, we find $T(D, L, N)$ directly as

$$T(D,L,N) = \frac{D^5L^3N}{1 - DL(1+L)N} \quad \text{(C6)}$$

Since we are not interested in the number of ones in the input bits, set $N = 1$ to get

$$T(D,L) = \frac{D^5L^3}{1 - DL(1+L)} \quad \text{(C7)}$$

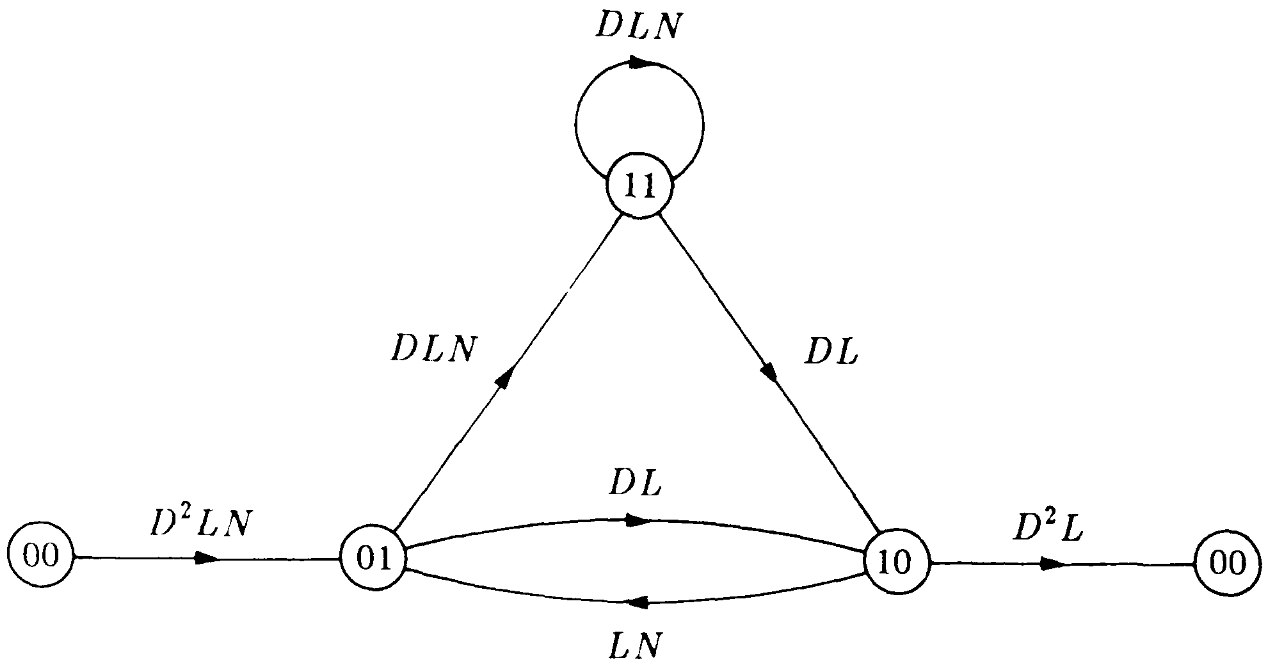

Figure C2. Modified state diagram.

Rate 1/2 k=3 code.

We define $a_i^{(\mathcal{L})}$ to be the sum of the coefficients of all terms of $T(D,L)$ for which the exponent of D is i and the exponent of L is less than or equal to $\mathcal{L}$. $a_i^{(\mathcal{L})}$ is the number of codewords of length less than or equal to $\mathcal{L}$ with i ones. The union bound on the probability that an error event occurs at bit $\mathcal{L}$ is

$$P_E^{(\mathcal{L})} \leq \sum_{i=1}^{m\mathcal{L}} a_i^{(\mathcal{L})} P_i \tag{C8}$$

This is overbounded by letting $\mathcal{L}$ go to infinity. We define $a_i = \lim_{\mathcal{L}\to\infty} a_i^{(\mathcal{L})}$, so

$$P_E \leq \sum_{i=1}^{\infty} a_i P_i \tag{C9}$$

for every bit. We find the coefficients a_i by evaluating $T(D,L)$ at $L=1$, since

$$T(D,L)|_{L=1} = \sum_{i=1}^{\infty} a_i D^i \tag{C10}$$

An additional simplification is realized by using the bound $P_i \leq (2\sqrt{p(1-p)})^i$. This yields

$$P_E \leq \sum_{i=1}^{\infty} a_i (2\sqrt{p(1-p)})^i = T(D)|_{D=2\sqrt{p(1-p)}} \tag{C11}$$

However, this is a very loose bound for large values of p. Because we are interested in the decoder performance up to about 10^{-2}, it is necessary to use the more exact expression of equation C9.

C.3 Numerical evaluation of Decoder Performance

Several steps were required in order to derive numerical results for the performance. First, the transfer function $T(D)$ had to be calculated. Next, the expression had to be

stated in the form of a power series expansion to find the coefficients a_i. Finally, the terms P_i had to be calculated and the summation of Eqn. C9 evaluated.

A computer program was written to give the equations relating the states of the modified state diagram for the rate 1/2 constraint length 7 code being modeled. The MACSYMA (©1976,1983 Massachusetts Institute of Technology, ©1983 Symbolics, Inc.) symbolic manipulation program was then used to solve the 64 simultaneous equations, giving an expression for $T(D)$ as the ratio of two polynomials. This expression is given in Table C1.

Table C1. Polynomial Ratio Solution to T(D)

Numerator$= -D^{80} + 16D^{76} - 120D^{72} - D^{70} + 562D^{68} + 8D^{66} - 1838D^{64} - 20D^{62} + 4429D^{60} - 18D^{58} - 8068D^{56} + 235D^{54} + 11218D^{52} - 678D^{50} - 11900D^{48} + 1097D^{46} + 9575D^{44} - 1094D^{42} - 5841D^{40} + 611D^{38} + 2795D^{36} - 49D^{34} - 1156D^{32} - 228D^{30} + 417D^{28} + 243D^{26} - 76D^{24} - 176D^{22} - 15D^{20} + 93D^{18} + D^{16} - 25D^{14} - 6D^{12} + 11D^{10}$

Denominator$= 2D^{76} - 32D^{72} + 240D^{68} + 3D^{66} - 1123D^{64} - 30D^{62} + 3662D^{60} + 131D^{58} - 8766D^{56} - 331D^{54} + 15763D^{52} + 561D^{50} - 21403D^{48} - 782D^{46} + 21746D^{44} + 1184D^{42} - 16133D^{40} - 1960D^{38} + 8344D^{36} + 2807D^{34} - 2751D^{32} - 3064D^{30} + 389D^{28} + 2509D^{26} + 267D^{24} - 1601D^{22} - 403D^{20} + 844D^{18} + 262D^{16} - 345D^{14} - 81D^{12} + 85D^{10} + 40D^{8} - 30D^{6} - 6D^{4} - 4D^{2} + 1$

The denominator was then expanded using the identity

$$\frac{1}{1-x} = 1 + x + x^2 + x^3 + \ldots \tag{C12}$$

The expansion was calculated out to 64 terms. This was then multiplied by the numerator, and the first 64 terms were retained. The coefficients are listed in Table C2.

We note that for large i, the number of codewords of Hamming weight i given by a_i is very large. For a low probability of symbol error $P_{e,symbol}$, P_i will be small enough to offset

Table C2. Coefficients a_i

i	a_i	i	a_i	i	a_i
10	11	54	$2.912797332 \times 10^{17}$	98	$1.243940152 \times 10^{34}$
12	38	56	$1.660510362 \times 10^{18}$	100	$7.091380930 \times 10^{34}$
14	193	58	$9.466139591 \times 10^{18}$	102	$4.042612776 \times 10^{35}$
16	1331	60	$5.396401324 \times 10^{19}$	104	$2.304588936 \times 10^{36}$
18	7275	62	$3.076348764 \times 10^{20}$	106	$1.313786523 \times 10^{37}$
20	40406	64	$1.753746820 \times 10^{21}$	108	$7.489557000 \times 10^{37}$
22	234969	66	$9.997656840 \times 10^{21}$	110	$4.269602640 \times 10^{38}$
24	1337714	68	$5.699405471 \times 10^{22}$	112	$2.433989993 \times 10^{39}$
26	7594819	70	$3.249083581 \times 10^{23}$	114	$1.387554717 \times 10^{40}$
28	43375588	72	$1.852218477 \times 10^{24}$	116	$7.910090375 \times 10^{40}$
30	247339453	74	$1.055901826 \times 10^{25}$	118	$4.509337848 \times 10^{41}$
32	1409277901	76	$6.019423091 \times 10^{25}$	120	$2.570656827 \times 10^{42}$
34	8034996288	78	$3.431517347 \times 10^{26}$	122	$1.465464930 \times 10^{43}$
36	$4.580875611 \times 10^{10}$	80	$1.956219246 \times 10^{27}$	124	$8.354236400 \times 10^{43}$
38	$2.611287754 \times 10^{11}$	82	$1.115189974 \times 10^{28}$	126	$4.762534003 \times 10^{44}$
40	$1.488634502 \times 10^{12}$	84	$6.357409478 \times 10^{28}$	128	$2.714997404 \times 10^{45}$
42	$8.486419243 \times 10^{12}$	86	$3.624194641 \times 10^{29}$	130	$1.547749762 \times 10^{46}$
44	$4.837861791 \times 10^{13}$	88	$2.066059586 \times 10^{30}$	132	$8.823320927 \times 10^{46}$
46	$2.757937903 \times 10^{14}$	90	$1.177807109 \times 10^{31}$	134	$5.029946958 \times 10^{47}$
48	$1.572231420 \times 10^{15}$	92	$6.714373564 \times 10^{31}$	136	$2.867442606 \times 10^{48}$
50	$8.962880896 \times 10^{15}$	94	$3.827690629 \times 10^{32}$		
52	$5.109505443 \times 10^{16}$	96	$2.182067383 \times 10^{33}$		

the coefficient a_i, so only the first few terms of the summation are significant. However, for $P_{e,symbol} = 0.04715$, which corresponds to $P_E = 10^{-2}$, the term $a_{138}P_{138}$ is equal to 2.16×10^{-4}, or about 2% of the total. For $P_{e,symbol} > 0.04715$, the power series yields a $P_E > 10^{-2}$. However, in this range, terms of the summation of order higher than 138 become significant. Because the numerical calculation truncates these terms, the power series result is not the true union bound. Of course, the bound becomes quite loose, so the partial sum is probably greater than the actual P_E. Nevertheless, we use the looser bound $P_E \leq 1$ for $P_{e,symbol} > 0.04715$.

Appendix D. Probability of Correct Packet Reception

D.1 Fixed Length Packets [TAIP84]

For a packet of length $\mathcal{L}$ being sent over a binary symmetric channel, the probability of the packet being correctly decoded, P_C, was bounded by Taiple [TAIP84] as follows. For correct decoding, we assume that the starting state is known and that a known tailer of $k-1$ bits is transmitted, so a total of $\mathcal{L}+k-1$ bits are encoded, and $m(\mathcal{L}+k-1)$ symbols are transmitted. Again, we choose the starting state and the tailer to be $k-1$ zeros. There are a total of $2^{\mathcal{L}}$ codewords which are candidates for comparison to the received sequence, since the decoder will only choose a codeword which starts and stops in the all zero state. We index the paths corresponding to these codewords as ξ_i. Let the event E_i be the event that path ξ_i is not chosen by the decoder. The packet will be correct if the path 0 is chosen by the decoder, or equivalently, if E_i does not occur for any $i > 0$. Thus,

$$P_C = \Pr\left(\bigcap_{i=1}^{2^{\mathcal{L}}-1} \mathrm{E}_i\right) \tag{D1}$$

Taiple [TAIP84] proves that for the events E_i,

$$P_C \geq \prod_{i=1}^{2^{\mathcal{L}}-1} \Pr(\mathrm{E}_i) \tag{D2}$$

We can group the non-zero codewords according to the first time at which they return to the zero state. Let B_x be the set $\{i|\xi_i$ returns to the zero state at bit $x\}$. The soonest a non-zero codeword can return to the zero state is in k bits, since a one followed by all zeros still must pass through all k stages of the encoder shift register. Thus, the first non-empty set B_x is B_k. The sets $\{B_x|k \leq x \leq \mathcal{L}+k-1\}$ form a partition of the indices $\{1, 2, \ldots, 2^{\mathcal{L}}-1\}$, so

$$P_C \geq \prod_{i=1}^{2^{\mathcal{L}}-1} \Pr(\mathrm{E}_i) = \prod_{x=k}^{\mathcal{L}+k-1} \prod_{i \in B_x} \Pr(\mathrm{E}_i) \tag{D3}$$

Let $\{\xi_i'\}$ be the set of paths starting at bit $-\infty$, passing through any arbitrary state at bit zero and ending in the zero state at bit $\mathcal{L}+k-1$. We can include in the product $\prod_{i \in B_x} \Pr(\mathrm{E}_i)$ the terms corresponding to paths which first return to the zero state at bit x but which did not start in the initial state of all zeros at bit 0 , and still retain the inequality of D3, since the probabilities of the extra terms are all less than 1 Let B_x' be the set $\{i|\xi_i'$ returns to the zero state at bit $x\}$. The set of all paths $\{\xi_i'|i \in B_x'\}$ are equivalent for all x, so we let $B' = B_x'$. The inequality of D3 now becomes

$$\begin{aligned} P_C &\geq \left(\prod_{i \in B'} \Pr(\mathrm{E}_i)\right)^{\mathcal{L}} \\ &\geq \left(\prod_{i \in B'} (1 - \Pr(\bar{\mathrm{E}}_i))\right)^{\mathcal{L}} \end{aligned} \tag{D4}$$

where $\bar{\mathrm{E}}_i$ is the compliment of E_i We multiply out the term $\prod_{i \in B'}(1 - \Pr(\bar{\mathrm{E}}_i))$ to get an infinite sum, which is lower bounded by the sum of lower order terms $1 - \sum_{i \in B'} \Pr(\bar{\mathrm{E}}_i)$. However, the event $\bar{\mathrm{E}}_i$ is the event that path ξ_i' is chosen as the correct path, and this is summed over all paths which are returning to the zero state. This sum is exactly the sum found in the union bound earlier. Thus,

$$P_C \geq \left(1 - \sum_{i \in B'} \Pr(\bar{\mathrm{E}}_i)\right)^{\mathcal{L}} = (1 - P_E)^{\mathcal{L}} \tag{D5}$$

D.2 Varying Symbol Error Rate

Using a similar approach, we can find an approximation to a bound for the probability of correct packet reception for a variable length packet being transmitted on a channel with varying symbol error rate, as in the network model. This approximation results in a model of decoder performance which is memoryless, and therefore can be incorporated into the Markovian network model.

The analysis of the Viterbi decoding algorithm assumes that infinite decoder memory is available, so that the chosen codeword will be the one which is closest to the received symbol stream. In a practical implementation, a finite memory must be chosen, and a sub-optimum choice must be made when the memory overflows. However, the memory can be made large enough that the probability of this occurring is negligible. Simulations have shown that a memory of approximately $5(k-1)$ information bits is sufficient for a rate 1/2, constraint length k decoder [CLAR81]. In the case of the $k = 7$ code, this is 30 bits.

Even if the packet length is not fixed, for a given packet of length $\mathcal{L}$ bits, the inequality of D3 is still valid. We again include those paths corresponding to codewords of length greater than x in the product, to give

$$P_C \geq \prod_{x=k}^{\mathcal{L}+k-1} \prod_{i \in B'_x} \Pr(\mathrm{E}_i) = \prod_{x=k}^{\mathcal{L}+k-1} \prod_{i \in B'_x} (1 - \Pr(\bar{\mathrm{E}}_i)) \tag{D6}$$

Also, the inner product can be lower bounded by the sum of lower order terms, so

$$\prod_{i \in B'_x} (1 - \Pr(\bar{\mathrm{E}}_i)) \geq 1 - \sum_{i \in B'_x} \Pr(\bar{\mathrm{E}}_i) \tag{D7}$$

We now partition B'_x into ${}^{+}B'_x$, those paths which diverged from the all zeros state within the last 30 bits, and ${}^{-}B'_x$, those paths which diverged more than 30 bits ago. The decoder memory is chosen such that

$$\sum_{i \in {}^{-}B'_x} \Pr(\bar{\mathrm{E}}_i) << \sum_{i \in {}^{+}B'_x} \Pr(\bar{\mathrm{E}}_i) \tag{D8}$$

Thus, we approximate the sum $\sum_{i \in B'_x} \Pr(\bar{\mathrm{E}}_i)$ by $\sum_{i \in {}^{+}B'_x} \Pr(\bar{\mathrm{E}}_i)$. To find $\Pr(\bar{\mathrm{E}}_i)$ for $i \in {}^{+}B'_x$, we must know the probability of symbol error during each of the previous 30 bits. However, the sum is monotonic in $P_{e,symbol}$. We can bound the sum by taking the highest and lowest

values of $P_{e,symbol}$ occurring in the last 30 bits and finding $\Pr(\bar{\mathrm{E}}_i)$ if all symbols had the extreme value of $P_{e,symbol}$.

We approximate the performance by using the current value of $P_{e,symbol}$ as the value experienced by the last 60 symbols. By using the bounds on $P_{e,symbol}$, we show in Section 4.C that this approximation is valid.

If the symbol error rate for all 60 symbols is $P_{e,symbol}(j)$, we can bound the sum by

$$\sum_{i \in + \mathcal{J}'_x} \Pr(\bar{\mathrm{E}}_i) \leq P_E(j) \tag{D9}$$

where $P_E(j)$ is the probability of first error given a symbol error rate of $P_{e,symbol}(j)$. If j_x is the state at bit x, using the memoryless approximation we find

$$P_C \gtrsim \prod_{x=k}^{\mathcal{L}+k-1} (1 - P_E(j_x)) \tag{D10}$$

Let $\mathcal{L}_j$ be the number of bits for which the state is j.

$$P_C \gtrsim \prod_{j=1}^{\infty} (1 - P_E(j_x))^{\mathcal{L}_j} \tag{D11}$$

As described in Appendix C, the decoder model overbounds P_E by $P_E \leq 1$ for those values of P_E greater than 10^{-2}. We use L to denote the state j such that $P_E(L) \leq 10^{-2}$ and $P_E(L+1) > 10^{-2}$. We bound P_C by zero if the state $L+1$ is reached. Thus, we only need to find P_C for those cases when $\mathcal{L}_j = 0$ for $j > L$. We then find

$$P_C \gtrsim \prod_{j=1}^{L} (1 - P_E(j_x))^{\mathcal{L}_j} \tag{D12}$$

Appendix E. Throughput Equation for the Half-duplex Model

As in Appendix A, we expand the state space to explicitly indicate the state of user i

We define the following:

t^* is the number of users transmitting not including user i

r^* is the number of users receiving not including user i

$$c_i \text{ is the state of user } i, \qquad c_i = \begin{cases} 0 & i \text{ is idle} \\ 1 & i \text{ is transmitting} \\ -1 & i \text{ is receiving} \end{cases}$$

$\mathcal{S}^*$ is the expanded state space,

$$\mathcal{S}^* = \left\{ (t^*, r^*, c_i) : \begin{array}{c} 0 \le t^* \le M-1 \\ 0 \le r^* \le t^* + c_i \\ t^* + r^* \le M-1 \\ c_i \in \{-1, 0, 1\} \end{array} \right\}$$

$\pi^*_{(t^*, r^*, c_i)}$ is the steady state probability of being in state (t^*, r^*, c_i)

$S(t^*, r^*, i)$ is the fraction of time that user i is successfully transmitting packets which found the network in state $(t^*, r^*, 0)$ upon arrival.

We again have $C_k(t^*, r^*, i)$ and $T_k(t^*, r^*, i)$ as in Appendix A.

$$E(C_k(t^*, r^*, i)) = \frac{1}{\lambda \pi^*_{(t^*, r^*, 0)}} \tag{E1}$$

The event $\{\mathcal{X}^* = (t^*, r^*, 0)\}$ is the same as the event $\{\mathcal{X} = (t, r) \text{ and } i \text{ is idle})\}$, $t = t^*$ and $r = r^*$, so

$$\pi^*_{(t^*, r^*, 0)} = \pi_{(t,r)} P_{SI}(t, r) \qquad \begin{array}{c} t = t^* \text{ and} \\ r = r^* \end{array} \tag{E2}$$

and

$$E(C_k(t^*, r^*, i)) = \frac{1}{\lambda \pi_{(t,r)} P_{SI}(t,r)} \qquad \begin{array}{c} t = t^* \text{ and} \\ r = r^* \end{array} \tag{E3}$$

By definition, the quantity $E(T_s|(t,r))$ is conditioned on the packet capturing the receiver. If the receiver is not captured, this corresponds to an immediate transition to the state Failure in the auxiliary Markov chain. Since $T_k(t^*, r^*, i)$ is not conditioned on capture, we have

$$E(T_k(t^*, r^*, i)) = E(T_s|(t,r)) \cdot \Pr(\text{receiver is captured } |t,r) \qquad \begin{array}{c} t = t^* \text{ and} \\ r = r^* \end{array} \tag{E4}$$

$\Pr(\text{receiver is captured } |t,r)$ is equal to $P_{DI}(t,r)\alpha_t$. Given these, we can find

$$\begin{aligned} S(t^*, r^*, i) &= \frac{E(T_k(t^*, r^*, i))}{E(C_k(t^*, r^*, i))} \\ &= \lambda \pi_{(t,r)} P_{SI}(t,r) P_{DI}(t,r) \alpha_t E(T_s|(t,r)) \qquad \begin{array}{c} t = t^* \text{ and} \\ r = r^* \end{array} \end{aligned} \tag{E5}$$

for those states $\{(t^*, r^*) : (t^*, r^*, c_i) \in S^*, \text{ for any } c_i\}$. Since $E(T_s|(t,r)) = 0$ for $t \geq L$, the throughput can be found by summing over the states S', so

$$S_i = \sum_{(t,r) \in S'} \lambda \pi_{(t,r)} P_{SI}(t,r) P_{DI}(t,r) \alpha_t E(T_s|(t,r)) \tag{E6}$$

Appendix F. Throughput Equation for Delay Models

F.1 Receivers Available for all Transmissions

The derivation is very similar to those of appendixes A and E.

We define the following:

w^* is the number of users in the holding state not including user i

t^* is the number of users transmitting not including user i

$$c_i \text{ is the state of user } i,\ c_i = \begin{cases} 0 & i \text{ is idle} \\ 1 & i \text{ is holding} \\ 2 & i \text{ is transmitting} \end{cases}$$

$$S^* = \left\{ (w^*, t^*, c_i) : \begin{array}{c} 0 \le w^* \le M-1 \\ 0 \le t^* \le M-1-w^* \\ 0 \le c_i \le 2 \end{array} \right\}$$

$\pi^*_{(w^*,t^*,c_i)}$ is the steady state probability of being in state (w^*, t^*, c_i)

$S(w^*, t^*, i)$ is the fraction of time that user i is successfully transmitting packets which left the holding state when the network was in state $(w^*, t^*, 1)$.

$$E(C_k(w^*, t^*, i)) = \frac{1}{\nu \pi^*_{(w^*,t^*,1)}} \tag{F1}$$

The event $\{\mathcal{X}^* = (w^*, t^*, 1)\}$ is the same as the event $\{\mathcal{X} = (w, t) \text{ and } i \text{ is holding})\}$, $w = w^*$ and $t = t^*$ so

$$\pi^*_{(w^*,t^*,1)} = \pi_{(w,t)} P_H(w,t) \qquad \begin{array}{c} w = w^* \text{ and} \\ t = t^* \end{array} \tag{F2}$$

which gives

$$E(C_k(w^*, t^*, i)) = \frac{1}{\nu\pi_{(w,t)}P_H(w,t)} \qquad \begin{array}{c} w = w^* \text{ and} \\ t = t^* \end{array} \tag{F3}$$

Also,

$$E(T_k(w^*, t^*, i)) = \alpha_t E(T_s|(w,t)) \qquad \begin{array}{c} w = w^* \text{ and} \\ t = t^* \end{array} \tag{F4}$$

Given these, we can find

$$\begin{aligned} S(w^*, t^*, i) &= \frac{E(T_k(w^*, t^*, i))}{E(C_k(w^*, t^*, i))} \\ &= \nu\pi_{(w,t)}P_H(w,t)\alpha_t E(T_s|(w,t)) \qquad \begin{array}{c} w = w^* \text{ and} \\ t = t^* \end{array} \end{aligned} \tag{F5}$$

and

$$S_i = \sum_{(w,t)\in S'} \nu\pi_{(w,t)}P_H(w,t)\alpha_t E(T_s|(w,t)) \tag{F6}$$

F.2 Half-duplex Model

We define the following:

w^* is the number of users in the holding state not including user i

t^* is the number of users transmitting not including user i

r^* is the number of users receiving not including user i

c_i is the state of user i, $\quad c_i = \begin{cases} 0 & i \text{ is idle} \\ 1 & i \text{ is holding} \\ 2 & i \text{ is transmitting} \\ 3 & i \text{ is receiving} \end{cases}$

$$a_i = \begin{cases} 0 & c_i = 0 \text{ or } 1 \\ 1 & c_i = 2 \\ -1 & c_i = 3 \end{cases}$$

$$S^* = \left\{ (w^*, t^*, r^*, c_i) : \begin{array}{c} 0 \le w^* \le M-1 \\ 0 \le t^* \le M-1-w^* \\ 0 \le r^* \le t^* + a_i \\ w^* + t^* + r^* \le M-1 \\ 0 \le c_i \le 3 \end{array} \right\}$$

$\pi^*_{(w^*,t^*,r^*,c_i)}$ is the steady state probability of being in state (w^*, t^*, r^*, c_i)

$S(w^*, t^*, r^*, i)$ is the fraction of time that user i is successfully transmitting packets which left the holding state when the network was in state $(w^*, t^*, r^*, 1)$.

$$E(C_k(w^*, t^*, r^*, i)) = \frac{1}{\nu \pi^*_{(w^*,t^*,r^*,1)}} \tag{F7}$$

The event $\{\mathcal{X}^* = (w^*, t^*, r^*, 1)$ is the same as the event $\{\mathcal{X} = (w, t, r)$ and i is holding$)\}$ $w = w^*$, $t = t^*$, and $r^* = r$, so

$$\pi^*_{(w^*,t^*,r^*,1)} = \pi_{(w,t,r)} P_H(w,t,r) \qquad \begin{array}{c} w = w^*,\ t = t^*, \\ \text{and } r = r^* \end{array} \tag{F8}$$

which gives

$$E(C_k(w^*, t^*, r^*, i)) = \frac{1}{\nu \pi_{(w,t,r)} P_H(w,t,r)} \qquad \begin{array}{c} w = w^*,\ t = t^*, \\ \text{and } r = r^* \end{array} \tag{F9}$$

Also,

$$E(T_k(w^*, t^*, r^*, i)) = P_{DI}(w,t,r)\alpha_t E(T_s|(t,r)) \qquad \begin{array}{c} w = w^*,\ t = t^*, \\ \text{and } r = r^* \end{array} \tag{F10}$$

Given these, we can find

$$\begin{aligned} S(w^*, t^*, r^*, i) &= \frac{E(T_k(w^*, t^*, r^*, i))}{E(C_k(w^*, t^*, r^*, i))} \\ &= \nu \pi_{(w,t,r)} P_H(w,t,r) P_{DI}(w,t,r)\alpha_t E(T_s|(w,t,r)) \quad \begin{array}{c} w = w^*,\ t = t^*, \\ \text{and } r = r^* \end{array} \end{aligned} \tag{F11}$$

and

$$S_i = \sum_{(w,t,r)\in S'} \nu \pi_{(w,t,r)} P_H(w,t,r) P_{DI}(w,t,r)\alpha_t E(T_s|(w,t,r)) \tag{F12}$$

References

[BELL80] S.Bellini and F. Borgonovo, "On the Throughput of an ALOHA Channel with Variable Length Packets", *IEEE Trans. Comm.* Vol. Com-28, No. 11, pp. 1932-1935, November 1980.

[BOOR80] R. Boorstyn and A. Kershenbaum, "Throughput Analysis of Multihop Packet Radio", *Proc. ICC '80*, Seattle, Washington, June 1980.

[BRAZ84] J. Brazio and F. Tobagi, "Theoretical Results in Throughput Analysis of Multihop Packet Radio Networks", *Proc. ICC '84*, Amsterdam, May 1984.

[CLAR81] G. Clark, Jr. and J. Cain, *Error-Correction Coding for Digital Communications*, Plenum Press, New York, 1981.

[FERG75] M. Ferguson, " A Study of Unslotted ALOHA with Arbitrary Message Lengths", *DATA COMM.*, P.Q., Canada, October 1975.

[KLEI75] L. Kleinrock, *Queueing Systems*, Vol. 1, Wiley, New York, 1975.

[MUSS82] J. Musser and J. Daigle, "Throughput Analysis of an Asynchronous Code Division Multiple Access (CDMA) System", *Proc ICC '82*, Philadelphia, June, 1982.

[PAPO65] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, McGraw-Hill, New York, 1965.

[PURS77] M. Pursley, "Performance Evaluation for Phase-Coded Spread-Spectrum Multiple Access Communication- Part I: System Analysis", *IEEE Trans. Comm.* Vol. Com-25, No. 8, pp. 795-799, August 1977.

[PURS83] M. Pursley, "Throughput of Frequency Hopped Spread-Spectrum Communications for Packet Radio Networks", *Proc. 1983 CISS*, John Hopkins Univ., March 1983.

[RAYC81] D. Raychaudhuri, "Performance Analysis of Random Access Packet-Switched Code Division Multiple Access Schemes", *IEEE Trans. Comm.* Vol. Com-29, No. 6, pp. 895-900, June 1981.

[TAIP84] D. Taiple, "Direct Sequence Spread-Spectrum Packet Radio Performance", *University of Illinois Coordinated Science Laboratory Report*, Report T-155, November 1984.

[VITE71] A. Viterbi, "Convolutional Codes and their Performance in Communication systems", *IEEE Trans. Comm.* Vol. Com-19, No. 5, pp. 751-772, October 1971.